Differential Techniques for the Accurate Estimation of Image Flow

by

Andrew S.L. Bainbridge-Smith, B.E.(Hons.)

Department of Electrical and Electronic Engineering

submitted in fulfilment of the requirements
for the degree of
Doctor of Philosophy

University of Tasmania, August 1996

Statement of Originality

This thesis contains no material which has been accepted for the award of any other degree or diploma in any tertiary institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person, except when due reference is made in the text.

Authority of Access

This thesis may be made available for loan and limited copying in accordance with the Australian Copyright Act 1968.

Andrew S. L. Bainbridge-Smith.

Abstract

One of the prime objectives of many robotic systems is the ability to judge depth for self navigation. This depth or range information in a three-dimensional scene has traditionally been obtained from point correspondences in static two-dimensional images, a process often known as photogrammetry. This approach has the major difficulty that point pairs corresponding to a common three-dimensional scene point are not easy to identify automatically in images. In more recent years a wide range of alternative approaches for estimating depth have emerged. One such group of techniques makes use of multiple images from a single moving sensor. Known as shape from motion algorithms, they often employ a three step process to estimating depth. The first step involves estimating the optical or image flow. The second step then estimates the global motion parameters of the image sensor from the image flow. The final step involves estimating the relative depth from the image flow and motion parameters. Thus in order to make reliable depth estimates, accurate image flow measures are required. This thesis, therefore, concentrates on the detailed analysis of techniques employed for accurately determining image flow.

Preliminary chapters introduce basic computer vision concepts and image flow algorithms. As some of the terminology used in this field is ambiguous, particular care has been placed on presenting a well defined set of terms, notably defining: "camera motion", the "motion field", "optical flow" and "image flow". Also defined is the "aperture problem" which describes a fundamental inability to perceive or measure motion in directions were no change in the image intensity function is present. It is demonstrated that all techniques rely on the presence of non-zero second order differential terms of the image sensor intensity function to overcome the aperture problem.

Following chapters present a review and comparison of a number of image flow techniques, with particular emphasis placed on gradient methods. It is argued that the fundamental problem of measuring image flow is to determine the velocity in as small an aperture as

possible. Based on this criteria it is concluded that the technique based on a weighted least squares solution of first order differential terms of the image intensity is the best generalized second order method for achieving this objective.

The practical issues of regularization of the gradient calculations and the influence of noise corruption are investigated. Conclusions drawn show the importance of smoothing and the effects of excessive smoothing, including a loss of spatial precision and an increase in numerical error due to the computation of small gradient terms. Importantly it is demonstrated that spatial and temporal gradients are not independent and consequentially their respective errors are correlated. It is thus shown that many image flow estimation techniques exhibit a systematic bias.

An investigation into methods of assessing the quality of image flow estimates is also made. Indicators of the quality of an estimate are required in order to evaluate the accuracy of subsequent estimates made from them; particularly depth estimates, segmentation of the image flow field, and determining the optimal aperture size for making image flow estimates. Findings illustrate that the correlation between many commonly used, or proposed, indicators of image flow error are poor.

Some investigative work into two innovative approaches for estimating image flow, based on the use of colour and actual camera motion parameters, is presented. It is shown that both colour and camera motion parameters provide an additional constraint to the estimation procedure, which can be used to improve the resolution of the image flow field by allowing smaller apertures to be considered. Preliminary results from these studies suggest that both sources of additional information improve the robustness of the algorithms.

The final part of this thesis presents some quantitative and qualitative results for image flow estimates from real imagery. A summary of conclusions is presented and a number of advanced issues and suggested extensions are outlined.

Acknowledgements

This thesis culminates four and a half years of thought, turmoil, torture and worry on more than my own behalf. To the many people who have suffered, and hopefully gained from this effort, I offer my heart-felt thanks. In particular, my extreme appreciation and thanks must go to my supervisor Dr Richard Lane who not only has provided much in the way of teaching, discussions and criticism but also tolerance for my ever lateness in getting things done.

A doctorate is a challenging journey, one I would not have undertaken if it were not for the initial encouragement of others. My most sincere thanks to my parents who have instilled in me an appreciation of the importance of learning, striving for the limits of ones talents, and a pinch of old-fashioned curiosity. Their encouragement and those of my siblings, Jeremy, David and Sarah have been undiminishing. My thanks also to Dr Bernard Guillemin at the University of Auckland, who in my last year at Auckland advised me honestly and wisely about the difficulties and rewards of undertaking such a programme.

I arrived in Hobart in March 1992 as an almost complete stranger. Though not a doctor some three years earlier, Dr David McLaren had kindly taken the time to talk to me about his experiences of doing an engineering degree. In Hobart as a doctoral student, he again took the time to talk to me about his enjoyment of doing a PhD, and also to help find accommodation, instill the proper study ethos and generally making me feel welcomed. To the other PhD students, particularly Dr Marc Stoksik, soon to be doctors Andrew Innes and Jason Pieloor go my appreciation for their help both academic and otherwise and for their unfaulting friendships. I shall always miss our morning coffee breaks, lunch-time indoor soccer, afternoon tennis matches and late evenings either working or playing computer games!

Thanks must also go to the staff and students at the Departments of Electrical and Electronic Engineering at the University of Tasmania and the University of Canterbury. In

particular to my associate supervisor Professor D.T. Nguyen, Tasmanians: Dr Jon Osborne and Glen Mayhew and Cantabrians: Associate Professor Peter Gough (for allowing me to use Canterbury's resources), Dr Phil Bones and Dr Adam Schwartz, and PhD students David Hawkins, Maria Luiza Lisboa, James Preddey and Volker Kuhlmann. This study would not have been possible with out the financial assistance of an Australian Postgraduate Award through the University of Tasmania, and the support of my parents and partner.

Finally and most importantly to my wife, Dr Ainsley Morris, go my thanks for her sacrifices, support and "carrot and stick" encouragement. Those who have been last shall now be first.

Contents

Preface

I arrived at the University of Tasmania in Hobart at the beginning of March 1992 to undertake a programme of study towards a PhD, having completed a bachelors degree in electrical and electronic engineering from the University of Auckland. My initial studies at the university concentrated on parameter estimation and least squares analysis which led into my first project concerning the training of neural networks. The initial hope of this work was to come up with a fast method of training a neural network for classifying Fourier descriptors of edge detected images. A conference paper on the training method was written and presented in Melbourne in early 1993. However, the result of this work was to spawn my interest in image processing rather than neural networks.

I had developed an interest in camera calibration, stereopsis and robotic vision. I quickly developed a new proposal: "measuring depth for an active vision system"; this has always been my long term ambition. Engineering, financial and interest constraints dictated that a single camera system be developed. Such a system leads to a three step process for estimating depth. The first step involves estimating image flow, the second global motion parameters of the camera, and the third step involves estimating relative depth from the image flow. In order to make reliable depth estimates, accurate image flow measures are required. With my youthful optimism I anticipated this first problem to be *solvable*, albeit restrictively, in two years leaving me a year to consider estimation of global motion parameters and depth. I would thus be finished by the time my research grant would expire! There is a saying about "best made plans coming to ruin" — things have not entirely gone to plan. In my case I am left with the sure knowledge that research almost always takes much longer than anticipated and that every brick of knowledge must be "hand-made with care." I thus present a thesis based on just over three years work on the problem of estimating image flow. Produced in the end with the kind financial support of my parents and partner.

In coming away from this work I hope that the reader will appreciate that although there is much published work in the field, many of the algorithms have a common base. In essence there are only two types of algorithm which can either be implemented in space-time or Fourier domains. This work concentrates mostly on differential techniques in the space-time domain and clearly shows that a local least squares method is best. Other results show the necessary requirements of regularizing derivative measures, the inadequacy of many published methods for indicating erroneous image flow measures, and significantly that the estimated image flow is likely to exhibit a systemic bias in its measure.

Chapters 5 and 6, although not consisting of a great proportion of this thesis, required a considerable effort. I easily spent some six months learning to control an industrial robot, becoming familiar with quaternion algebra, calibrating cameras and other implementation requirements. In some respects I found this work very rewarding; it was practical. However, in the end, mechanical jitter, instability and poor accuracy of the robot system meant that much of the data obtained from this source was unusable. Vision systems are, as of yet, not robust enough to handle pixel jitter in the order of 6 pixels/frame! There is much yet to be done in this exciting field, and I hope the reader is left with the same level of excitement that I had in developing and writing this work.

Thesis Organization

This thesis is organized into seven chapters. The first chapter gives an introduction to the field of robotic vision and the role of image flow. Chapter 2 outlines terminology and computer vision concepts, as well as relevant practical issues. Chapter 3 presents a review of various image flow algorithms, with a comparison between correlation and differential based methods. Chapter 4 provides an in-depth analysis and comparative work of differential methods of estimating image flow. Also presented is an investigation into published methods of measuring the accuracy of image flow estimates. Chapter 5 offers some innovative possibilities afforded by the use of colour imagery, as opposed to the traditional use of greyscale. Also considered is the use of camera global motion parameters when these are known. Chapter 6 examines both quantitative and qualitative results from real image sequences. Finally, Chapter 7 presents the overall conclusions, a summary of the contributions and the major results of this thesis. This chapter also concludes with suggested extensions of this work for future research.

Supporting Publications

This research has resulted in two journal papers and a number of conference publications. These are listed below:

- A. Bainbridge-Smith, M.A. Stoksik and R.G. Lane, "Optimization of Multi-layer Neural Networks using Gauss-Newton Minimization", in the Proceedings of the Fourth Australian Conference on Neural Networks (ACNN '93), Melbourne, Australia, pp.114—117, Feb. 1993

- A. Bainbridge-Smith and R.G. Lane, "Incorporating the Aperture Problem into Optical Flow Measurement", in the Proceedings of the First New Zealand Conference on Image and Vision Computing (IVCNZ '93), Auckland, New Zealand, pp.431—438, Aug. 1993

- A. Bainbridge-Smith and R.G. Lane, "Statistically Based Computation of Optical Flow", in the Proceedings of DICTA-93, Australian Pattern Recognition Society, Sydney, Australia, pp.228—235, Dec. 1993

- R.G. Lane, N.F. Law and A. Bainbridge-Smith, "Ensemble Deconvolution using a Wavefront Sensor", in the Proceedings of DICTA-93, Australian Pattern Recognition Society, Sydney, Australia, pp.236—243, Dec. 1993

- A. Bainbridge-Smith and R.G. Lane, "Measuring Image Flow from Colour Imagery", in the Proceedings of Electronic Colour Imaging and Applications Workshop (DICTA-94), Australian Pattern Recognition Society, Canberra, Australia, pp.9—14, Dec. 1994

- A. Bainbridge-Smith and R.G. Lane, "Generalised Differential Optical Flow", in the Proceedings of IVCNZ-95 Workshop, Lincoln, New Zealand, pp.113—118, Aug. 1995

- A. Bainbridge-Smith and R.G. Lane, "Consistent Optical Flow", in the Proceedings of DICTA-95, Australian Pattern Recognition Society, Brisbane, Australia, pp.673—678, Dec. 1995

- A. Bainbridge-Smith and R.G. Lane, "Measuring Confidence in Optical Flow Estimation", *IEE Electronic Letters*, **32**(10), pp.882—884, May 1996.

- A. Bainbridge-Smith and R.G. Lane "Determining Optical Flow Using a Differential Method", *Image and Vision Computing*, **15**(1), pp.11—22, January 1997.

Glossary

Symbols and Abbreviations

$\odot$	Convolution operator
$\otimes$	Quaternion multiplication operator
α	Spatiotemporal smoothing coefficient
β	Window size and weighting coefficient for local first order gradient technique (IF5)
γ	Weighting coefficient for Horn and Schunck's (IF4) smoothness constraint
λ	Weighting coefficient for the augmented second order gradient technique (IF6)
IF1:	Singh and Allen's original correlation method of computing image flow.
IF2:	A variant of Singh and Allen's correlation method.
IF3:	A variant of Singh and Allen's correlation method.
IF4:	Horn and Schunck's differential method of computing image flow.
IF5:	Local first order differential method of computing image flow.
IF6:	Local augmented second order method of computing image flow.
IF7:	Local second order method of computing image flow.
IF8:	A variant of method IF4 that incorporates colour information.
IF9:	A variant of method IF5 that incorporates colour information.
IF10:	A variant of method IF5 that incorporates known camera motion parameters.

Terms

- *Motion Parameters:*
  Parameters, 6, that describe the complete motion of a body in 3D space — 3 positional, 3 orientation.

- *Camera Motion or Egomotion:*
  Movement of the camera relative to the scene according to its motion parameters. Egomotion means self motion.

- *Motion Field:*
  Scene points move with velocities in 3D space relative to the camera as a result of camera motion, this collection of velocities is called the motion field.

- *Optical Flow:*
  The perspective transformation of the 3D motion field velocities into 2D space defines a field of 2D velocities. Optical flow is thus geometrically related to the motion field. Depending on context the optical flow may refer to the entire field or a point within the field.

- *Image Flow:*
  An estimate of the optical flow made from a sequence of images. This is a calculated quantity made from measured data and hence only approximates the optical flow. Depending on context this may refer to the entire field or a point within the field.

- *Aperture Problem:*
  The inability to observe or measure all components of the optical flow, especially in directions parallel to the spatial gradients of the image intensity.

- *Vernier Flow:*
  A field of velocities, or depending on context a point within a field, where the velocity at a point is only known in one direction (vernier direction). This direction is usually parallel to the spatial gradient of the image intensity, and arises from measurements made from data suffering from the aperture problem.

- *Needle diagram, Flow field diagram:*
  A diagram illustrating an optical, image or vernier flow field. Each vector shows both direction and magnitude of the movement of a picture element.

- *BRDF:*
  Bidirectional reflectance distribution function (BRDF), a model which describes the physical reflectance property of a material.

- *Chromaticity:*
  A term that collectively describes the two colour components hue and saturation. These describe the colour but not the intensity of light.

- *Achromatic:*
  Lacking colour — the intensity or greyscale component of light.

Chapter 1

Introduction

The rapid development in the mid 1970's of robotics has in part been due to the advent of cheaper computing power made possible by integrated circuitry. This has seen a progression from simple automated manufacturing techniques, such as those initially used in mass car production, to sophisticated production lines which require substantially less human input within the actual production cycle. Robots are well suited to performing the repetitive tasks which are common-place in most production cycles, as their "concentration" and "attention" never wane from the programmed job. From an engineering point of view, robotics offer multi-functional machines that can rapidly change tasks, often within minutes or even seconds. This results in a flexible, programmable production line. These production lines can be "re-programmed" within weeks instead of the months required in many "hard-wired" or fixed-automated environments, or alternatively, those simple and totally non-automated environments.

Popular science fiction, and much of the public perspective, have viewed robots as intelligent machines with human features; bipeds with two arms with 5 fingered hands, and a head with eyes, ears, nose and a mouth. Isaac Asimov, is one of the principal authors who has helped foster this public perception through his fictional work on robots and computing [33]. The word robot is derived from the Czech *robota,* meaning compulsory labour or "to work" in other Slavic languages [41]. Its modern sense stems from the Czech playwright Karel Ĉapek and his work "R.U.R." (Rossum's Universal Robots).

The mechanics of a human-like machine are extremely complex, research is still attempting to develop a hand of similar dexterity, strength and sense of touch as our own human hands. Additional to the mechanical issues are the sensory requirements of sight, hearing, smell, taste and touch, as well as control and other knowledge requirements. Asimov's robots were adept at performing "simple" tasks such as domestic and cleaning duties, hazardous tasks and the mundane, even within the same environment as their human supervisors.

These tasks are, however, anything but simple. The vision process alone poses one of the most difficult because of the large volume of information that must be processed. One second of a standard TV signal consists of approximately 6MB of data [35, 36, 38, 42]. It is therefore not unreasonable to question the need to research and develop machinery that mimics human functions. In order to answer this question we must clearly understand our expectations from such research. From a medical perspective much of the mechanical work is valuable in terms of prosthetics. However, from an engineering or manufacturing standpoint such value is unlikely. Nonetheless, much of the research into sensory functions could prove useful, for example the use of vision in object recognition and navigation. It is, in fact, the lack of these additional sensory abilities that has seen a limiting in the interest in robotics by industry. Hence, although we are likely to see the development of many of the functional units required to develop a human-looking robot, it is less likely that these units will be put together to build such a machine.

While current machines are orders of magnitude more flexible task wise than their predecessor, they are nevertheless still limited to fixed programmed movements and are largely oblivious to their surrounding environment. For robotics to move forward we must add sensory abilities to our machines. Indeed they are required in order to develop such systems as industrial carpet cleaners, automated warehouse storage, high-tech surveillance and so on, as industry and consumers demand.

1.1 Robotic Vision

Vision plays a significant role in many of our interactions with the rest of the world. Without it we would be reliant mostly on touch to identify the objects about us. Touch, however, has a number of drawbacks. Being primarily a mechanical process it is much slower in acquiring a "mental picture" of its surroundings. Moreover, not all objects of interest maybe within the range of touch and not all objects are touchable; either through a lack of touch, because

doing so is unsafe, or touch will lead to an undesirable change in the world.

Take as an example, a situation of being placed in a vast expanse of flat grassland in which a wandering pride of hungry lions is also located. Early identification of the pride is essential in order to take evasive action. Touch is not possible from an issue of safety, and because vision is non-invasive to the environment, retreat without giving away your location is possible. There is much to be said for letting sleeping lions lie!

Human interest in vision, has extended from the hunter/gather/survival requirements, which we rather take for granted, into fields of scientific and engineering endeavour. Initially this was through astronomy which aided farming and navigation. Vision was looked upon as a useful tool in this regard, although it was not until the likes of Newton etc., that an interest in light and the vision process started to develop [1]. Interestingly, a comprehensive connection between the vision process and geometry was only formed in the mid 1800's [2, 37]. With the advent of the camera came surveying from photographs, [1, 2]. Called photogrammetry the field quickly developed and these same techniques form the cornerstone of many machine vision algorithms [1, 2, 10, 14].

Machine vision looks at solving the problem of making value judgements from the recognition of objects within the image, and, or the measurement of features within the image. These tasks we perform "effortlessly" with our 10^{12} odd neurons [37]. Despite the development of ever increasingly computationally powerful computers, these computers are nevertheless no where near the ability of even emulating the common house fly in its ability to navigate a hazardous environment. This is of course a difficult problem. Fortunately some success has been achieved for much simpler tasks, such as self navigating laboratory robots where the environment is highly controlled. This has been driven by strong research evidence in neurobiology, neuroanatomy, psychophysics, and psychology that suggest cues such as shading (image intensity variation), texture (distribution of surface markings), contours and line-drawings, motion and stereo are very helpful in deducing properties of three-dimensional surfaces from their visual images [49–51, 74, 84, 94, 95, 116, 150, 173, 187, 188, 208, 216, 217].

One of the prime objectives of many robotic systems is to emulate the ability of animals in self navigation. Judging depth, the distance between the camera and points in the scene, is crucial to achieving this goal. Range information in a three-dimensional scene has traditionally been obtained from point correspondences in static two-dimensional images,

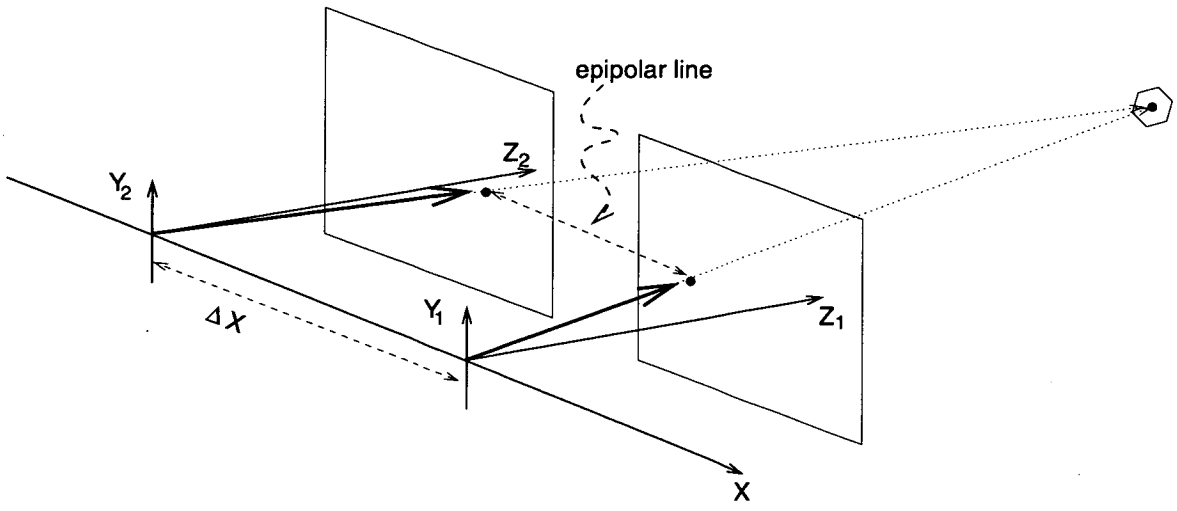


Figure 1.1: *Model of the binocular vision process; a scene point is imaged onto two separate sensors separated by a distance ΔX. The two image points line on the epipolar line which runs parallel to the X axis. The difference of the ordinate values of the imaged points from their respective axes is related to the distance of the object and the sensors separation.*

a process which has been utilized for many years by surveyors. The technique relies on locating common points in the three-dimensional scene from images taken from multiple separate cameras or a moving camera over time. It is the same binocular vision process by which many animals, ourselves included, see. Figure 1.1 illustrates this process. Comparing the *disparity* or difference between the scene points position in the two views, and given the separation of the two sensors, the depth can then be estimated by the use of triangulation. A major difficulty with automating this approach is that the point pairs, corresponding to a common three-dimensional scene point, are not easy to identify in images which are formed from significantly different viewpoints. Large viewpoints allow the possibility of multiple matches of candidate point pairs, a difficulty known as the correspondence problem which continues to receive wide attention from researchers. For reliable identification of point correspondences the viewpoints of the images need to be quite close, i.e. the angle of separation is small. Unfortunately, this is fundamentally in conflict with the need for accuracy in depth estimates which requires the images to be taken from widely spaced viewpoints to enable accurate triangulation [1, 2, 10, 14].

1.2 Visual Motion

In more recent years a wide range of new approaches for estimating depth have emerged, including: shape from shading, shape from texture, depth from focus, and shape from motion [45,49,53,59,68,86,92,101,108,117,125,128–131,133,141,142,160–162,168–171,177, 179,196,202,203,209,218,221,222]. In the first two cases, depth information is inferred from a single image only. Shape from motion and depth from focus algorithms differ in that they make use of multiple images. Depth from focus can be considered, however, as a special case of shape from motion, as it utilizes the disparity between two images caused by changing the focal length and knowledge of the camera's intrinsic parameters to estimate depth. Provided the motion parameters of the camera are known and the intrinsic parameters of the camera are not changed, shape from motion algorithms, which use just two images, are equivalent to measuring a binocular image pair. Other motion based algorithms, which make use of more than two images, can be thought of as binocular vision techniques enhanced by the ability to track points over long sequences.

Motion analysis is most commonly performed on an image sequence obtained from a single camera, although some approaches based on the motion of stereo cameras have also been considered [83,137,159,218,219]. It is the monocular vision approach that is adopted in this work, due to its simplicity and cost effectiveness (utilizing only one camera). An example of a monocular image sequence is shown in Figure 1.2. This is the "Hamburg Taxi" image sequence which is one of a number of freely available standard image sequences. Appendix A illustrates examples of all the image sequences used in this thesis, along with information about where they may be obtained.

Techniques for computing visual motion are based on a comparison of adjacent temporal frames. The simplest comparison method is the difference operator. Unfortunately taking the difference of two images alone is insufficient to map the displacement of a picture element from one frame to the next, as it only provides an indication that a change has taken place. Figure 1.3 illustrates two frames from the Hamburg Taxi sequence, circles have been placed around the four principal objects in motion. An absolute difference of the two frames is shown in Figure 1.4 where darker pixel colours indicate a larger difference or change. Change due to the motion of the three cars is clearly visible, however the velocities of the cars in pixels per frame is not computed. Additional problems arise from signal noise, as evidenced by the poor discrimination of the pedestrian's motion in the upper left hand corner and the non-zero differences computed in the largely stationary background. A more fundamental

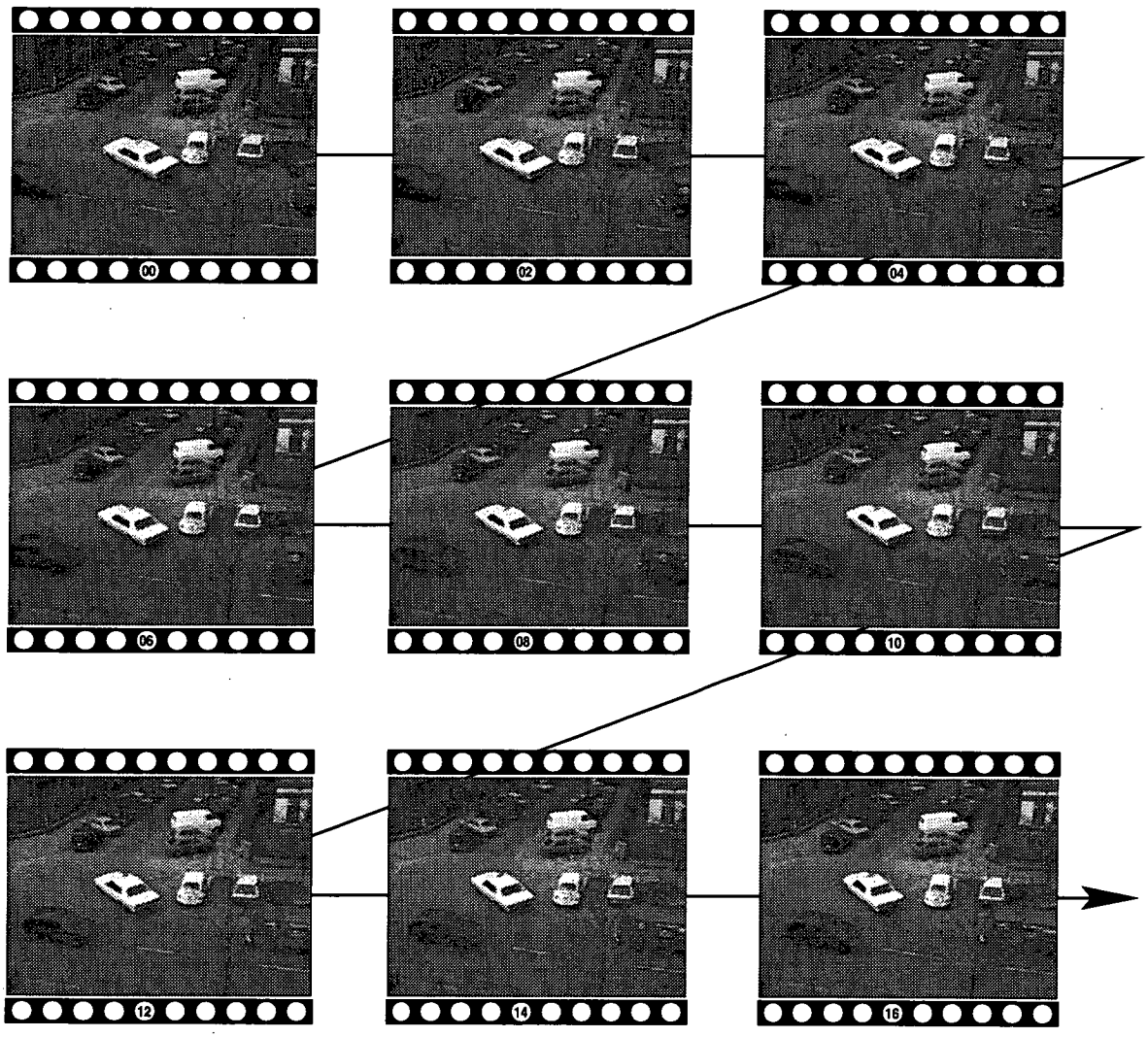


Figure 1.2: *A sequence of images from the "Hamburg Taxi" sequence.*

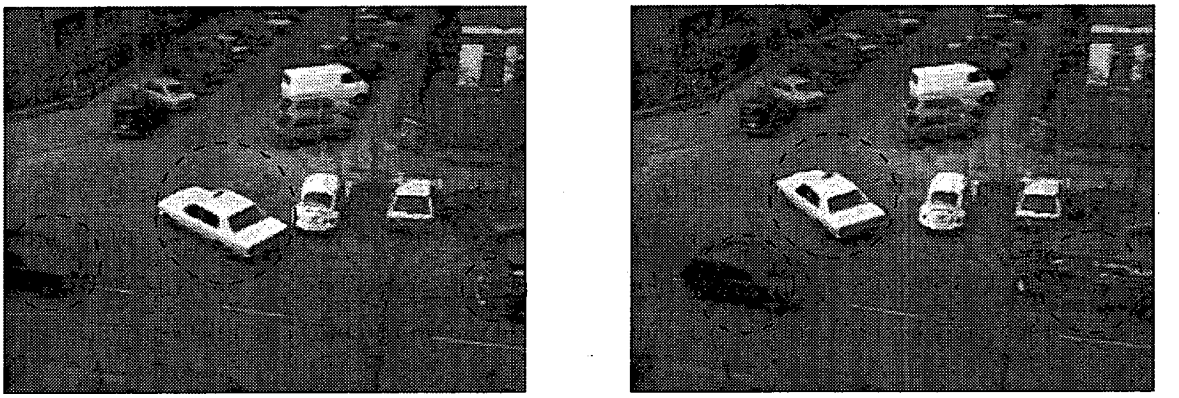


Figure 1.3: *Circles highlight the movement of objects from the first frame (left) to those of a later (right).*

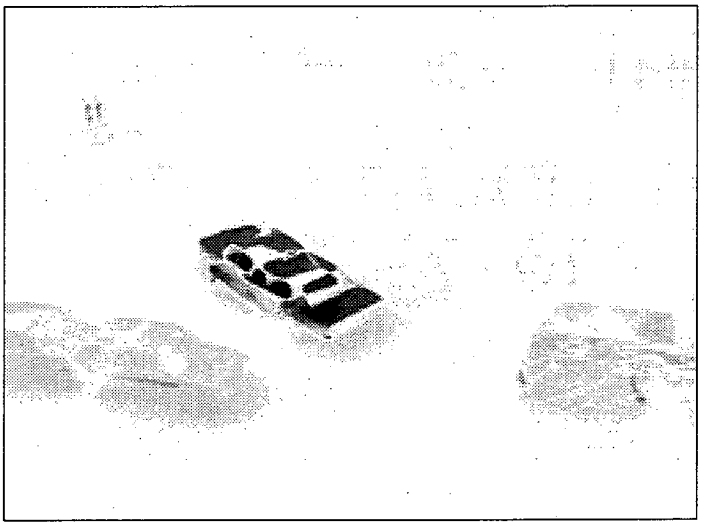


Figure 1.4: *Taking a difference image of the two frames shown in Figure 1.3 shows that the circled objects have moved.*

limitation is that this method is only suitable for situations where the camera is stationary and the background constant.

What is required is a much more sophisticated solution where the displacement from one frame to the next is clearly shown. Figure 1.5 shows an example of one such estimate. It is often referred to as a "needle diagram" or "flow field diagram", where each of the vectors shows the direction and magnitude of the movement of a picture element. Such a diagram

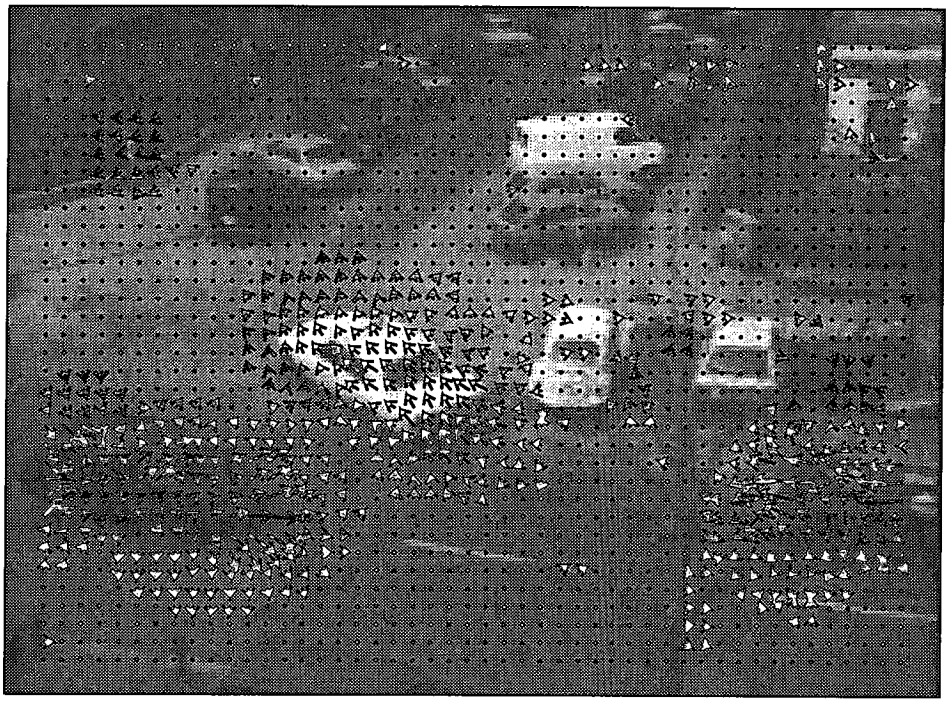


Figure 1.5: *A needle diagram showing the movement of picture elements in time. This flow field was computed about the 9^{th} frame in the Hamburg Taxi sequence using the author's implementation of Horn and Schunck's image flow algorithm.*

can be constructed by performing a point-wise match of each pixel in one image to a pixel in the other. However this is a nightmare computationally, with a potential of N^2 matches, where N is the total number of pixels per frame. Furthermore, if pixel intensity is the sole basis of comparison, then the algorithm is likely to suffer from many equally likely matches.

Two more needle diagrams are shown in Figures 1.6 and 1.7. These are computed for synthetic image sequences which were created from the simulated motion of a camera looking at a tree. Like the estimate for the Hamburg Taxi sequence they are computed using Horn and Schunck's algorithm [102] which is presented later in this thesis, along with other algorithms that provide computational superior alternatives to the point-matching approach. These diagrams also illustrate another feature of the algorithms to be presented, the ability to compute a dense flow field.

Success in measuring image displacement, referred to as image flow, will inevitably give rise to accurate and valuable relative depth estimates. This in turn allows confident decisions to

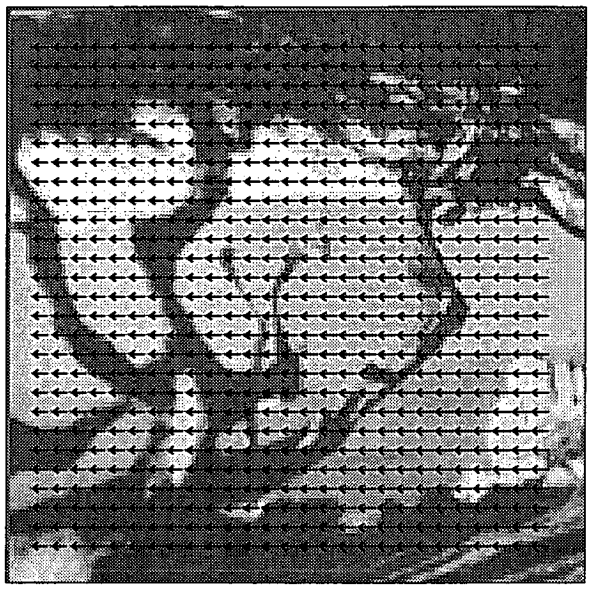


Figure 1.6: *Estimating the image flow for frame 20 from the Translating Tree sequence using Horn and Schunck's method.*

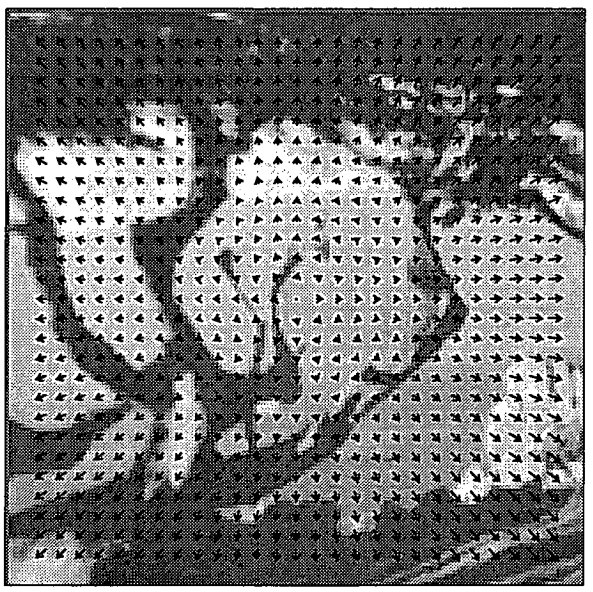


Figure 1.7: *Estimating the image flow for frame 20 from the Diverging Tree sequence using Horn and Schunck's method.*

be made about how to interact with the world. In a robotics environment such decisions form the basis of controlled motion, and thus it seems eminently sensible to use the information gained during movement to refine the control process. Vision therefore provides a potential feedback process to motion control. The key to success of the feedback process is the accurate measurement of image flow.

Although the approach taken and developed in this thesis is designed for robotic vision, the measurement of image flow has successfully found commercial application in such fields as image compression for High Definition Television (HDTV), and the estimation of wind speed and ocean currents for meteorological forecasts and climate studies [3,140,164,174,180,230]. Other applications include measuring plant growth, tracking cataracts and flight-systems to assist pilots [57,98,189]. As in the case of robotics, dense flow fields are required along with some indication of the reliability of each measure. Resolution, density, accuracy and reliability of image flow are therefore the prime factors considered when comparing the performance of various techniques.

1.3 Thesis Structure

This thesis presents various techniques for the accurate estimation of image flow. The topic is presented in a manner that develops these techniques for inclusion in an active vision system. The thesis is written in a style that groups together common topics into single chapters, with accompanied relative experimental work. However, only brief conclusions are drawn in these chapters as a single comprehensive conclusion is left to the last chapter. This allows a discussion of common results which span more than one chapter.

Firstly, a discussion of image formation and camera motion is made at the beginning of Chapter 2. A formal definition of image disparity, called optical or image flow, is then presented along with the fundamental obstacle faced by all algorithms based on measuring image disparity, namely the aperture problem. Finally, Chapter 2 presents prerequisite knowledge on the numerical techniques required by practical systems as well as the method by which errors in the disparity estimates can be measured.

Chapter 3 presents a review of the various classes of image flow estimators. It is demonstrated that in essence there are only two estimator types: correlators and differential methods. The first are conceptually easier to understand, while the latter are more compu-

tationally efficient. A brief comparison of algorithms from both of the two classes is made on a number of image sequences. The chapter concludes with a detailed investigation of Horn and Schunck's differential method with comparisons to results obtained by Barron et al.[54] for their implementation of this algorithm. Such analysis and comparisons are essential in order that any enhancements made are meaningful and addresses the criticism;

> "Computer Vision suffers from an overload of written information but a dearth of good evaluations and comparisons."
>
> ——————— Price[175]

It is fortunate that in the field of image flow analysis some recent efforts to address this criticism exist, [54–56, 167, 226], especially the work of Barron and others at the University of Western Ontario. Unfortunately, some of these results were in error, in particular their implementation of Horn and Schunck's algorithm. The results presented in this thesis therefore substantially differ from those of Barron et al., [52, 132].

Chapter 4 extends the detail of this comparative work, concentrating on the class of differential or gradient based techniques for measuring image flow. Comparisons are made with respect to the resolution, density and accuracy of the image flow. Noise analysis and estimator reliability are also investigated since, these are crucial to estimating the reliability of depth estimates.

In Chapter 5 investigations of two innovations for improving flow estimates are made. Both relate to the inclusion of additional information. The first considers the effect of colour information. The cost of "off-the-shelf" colour cameras is comparable to many greyscale (monochrome or black-and-white) cameras, making the purchase of this style of sensor attractive to a number of researchers. The second considers the affect of assuming that the camera motion is known a priori. This is not an unrealistic assumption as the control algorithm moving the camera can, at least in part, supply this information. Inevitably though this information will be partially erroneous, due to noise and mechanical effects, such as slippage, which give rise to inaccurate measurements. The effects of these errors are briefly investigated.

The performance of various algorithms on realistic image sequences is analyzed in Chapter 6. Quantitative analysis of the type presented in preceding chapters is used. Unfortu-

nately, there is a lack of suitable real image sequences which have known camera motion. Consequently many of the results presented in this chapter are also of a qualitative nature.

Finally, a summary together with a number of points of interest are discussed in the conclusions of Chapter 7. Also included in this chapter is a brief discussion of possible future extensions of this work, plus other advanced issues and points of interest that connect this research to that made in other fields. This includes such topics as path planning for navigation, time to collision and segmentation of the motion field.

Chapter 2

Background

This chapter introduces the concepts of image formation, camera motion, optical and image flow, and related practical issues. The objective is to clearly define the terminology used and in this regard we make a clear distinction in the use of the terms optical and image flow. The former is a geometric term that is exact and independent of the image formation process, while the later is an estimated quantity resulting from noisy image measurements. In clarifying the distinction between these terms, it is necessary to develop an understanding of the conditions which affect the accuracy of measuring moving picture elements. The most significant difficulty faced by any image flow algorithm is over-coming the aperture problem. A discussion of the sampling theorem for moving pictures, numerical techniques and other practical requirements demonstrates the effects of some of the principal simplifying assumptions. Finally, an important discussion on error metrics is made, as in reality an estimate of unknown accuracy is only marginally more useful than no estimate at all.

2.1 Image Formation

The process of image formation is crucial in order to develop a greater understanding of the process of how pictures change in time. Two distinctive processes are involved in the image formation process: how energy is radiated from a scene, and how this energy is measured.

Measuring Process

The measurement process involves collecting the energy emitted from the scene on an appropriate sensor. In machine vision problems, the sensor is typically a camera based on a two-dimensional CCD array. Consequently, the image plane of the camera is modelled as a set of cells mapped out on a rectangular grid (see Figure 2.1) where each cell measures the amount of energy it receives over a period of time. The measured photon count in each cell is then digitized for further processing by a computer system.

The correspondence between a point in the scene and its image on the image plane depends on the relative position of the scene point to the camera sensor, and the intermediate optics of the camera lens. The mapping between scene and image points is defined by the perspective transform,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{fX}{Z} \\ \frac{fY}{Z} \end{bmatrix}, \tag{2.1}$$

where uppercase letters, (X, Y, Z), denote the three-dimensional coordinates of scene points and lowercase letters, (x, y), denote the two-dimensional coordinates of image points. All light rays incident on the image plane must pass through the focal point which is at a distance f from the plane, Figure 2.2. The number of photons exciting the cell is dependent on the position of the scene relative to the sensor, the attitude and reflectance properties

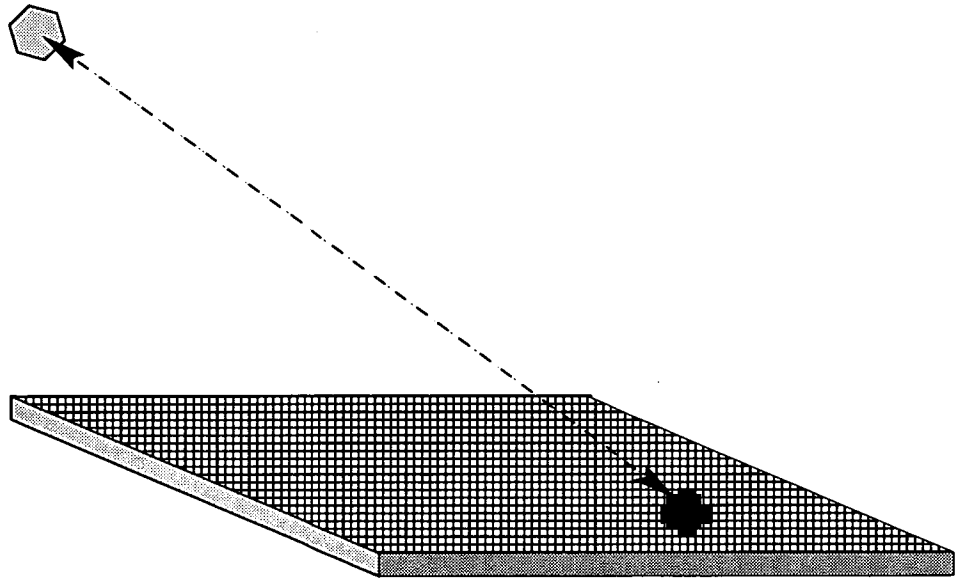


Figure 2.1: *A CDD array is a sensor made up of cells laid out on a rectangular grid.*

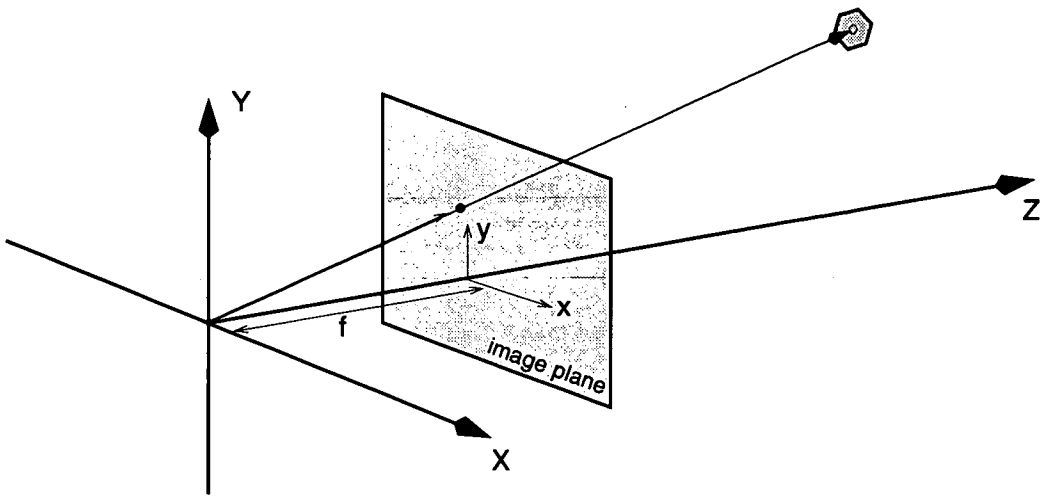


Figure 2.2: *Scene points in the three-dimensional coordinate system* (X, Y, Z) *are projected onto the two-dimensional sensor or image plane* (x, y) *at a distance* f *from the origin. The two coordinate systems are related by the perspective projection.*

of points in the scene, and the distribution of light sources.

Radiometry

The amount of light falling on a surface is called the *irradiance*, and is measured in power per unit area (Wm^{-2}). The amount of light radiated from a surface is called the *radiance*, and is measured as the amount of power per unit area of unit solid angle ($Wm^{-2}\text{rad}^{-1}$). The radiance measure is complicated because the source can radiate light into many differing directions with varying amounts of energy. In order to define this we need to compute the light radiated in a given solid angle, where the solid angle is defined as an angle equal to the area cut out by a cone on the unit sphere. Figure 2.3 shows a small planar patch of area A at a distance R from the origin with an attitude angle of ϕ. The foreshortened area, or effective area oriented towards the origin (illustrated by the lighter grey region), is equal to $A\cos\phi$ and thus subtends a solid angle,

$$\Theta = \frac{A\cos\phi}{R^2}, \tag{2.2}$$

of the unit sphere.

In Figure 2.4 we see that the light radiating from a scene patch δO with an attitude ϕ to the centre of the lens has an effective area of $\delta O\cos\phi$ towards the lens. This light is concentrated on a cell with area S in the image plane. If the radiance of the object patch

in the direction of ϕ is given by L then the power emitted by the patch is,

$$\delta P = L\delta O \cos(\phi)\, \Omega, \tag{2.3}$$

where Ω is the solid angle subtended by the lens as seen by the object and represents the light catching potential of the lens for the scene patch, Figure 2.5. The lens has an area $\frac{\pi}{4}d^2$ with an attitude of α to the scene patch. Given that the distance between lens and object is $Z/\cos\alpha$, then by substitution of these relationships into Eq.(2.2) the solid angle is defined by,

$$\Omega = \frac{\frac{\pi}{4}d^2 \cos\alpha}{(Z/\cos\alpha)^2}. \tag{2.4}$$

Since this power is concentrated in the image and assuming no losses in the lens, the irradiance measured by the cell will be,

$$I = \frac{\delta P}{S} = L\frac{\delta O}{S}\cos(\phi)\,\Omega. \tag{2.5}$$

The solid angle subtended by the image plane cell, S, to the lens is equal to the solid angle subtended by the object patch, δO, to the lens, as evident in Figure 2.4. Hence,

$$\frac{S\cos\alpha}{(f/\cos\alpha)^2} = \frac{\delta O\cos\phi}{(Z/\cos\alpha)^2}$$

$$\Longrightarrow \qquad \frac{\delta O}{S} = \frac{\cos\alpha}{\cos\phi}\left(\frac{Z}{f}\right)^2. \tag{2.6}$$

By substituting Eq.(2.6) into Eq.(2.5) the irradiance of the measuring cell is given as,

$$\begin{aligned} I &= L\cos(\alpha)\left(\frac{Z}{f}\right)^2 \Omega \\ &= \tfrac{\pi}{4}L\cos^4(\alpha)\left(\tfrac{d}{f}\right)^2. \end{aligned} \tag{2.7}$$

Since the angle of view, α, is typically narrow, Eq.(2.7) is insensitive to the fourth power cosine term. Moreover other factors, such as vignetting effects (power loss in a lens system, see [7]), are usually more significant, and in practice the fourth power cosine term can usually be ignored. We thus note that the irradiance measure is proportional to the scene radiance and the square of the reciprocal f-number (f/d) .

The scene radiance L, however, depends not only on the amount of light falling on the surface but also the fraction and distribution pattern of its reflection. A mirror is an

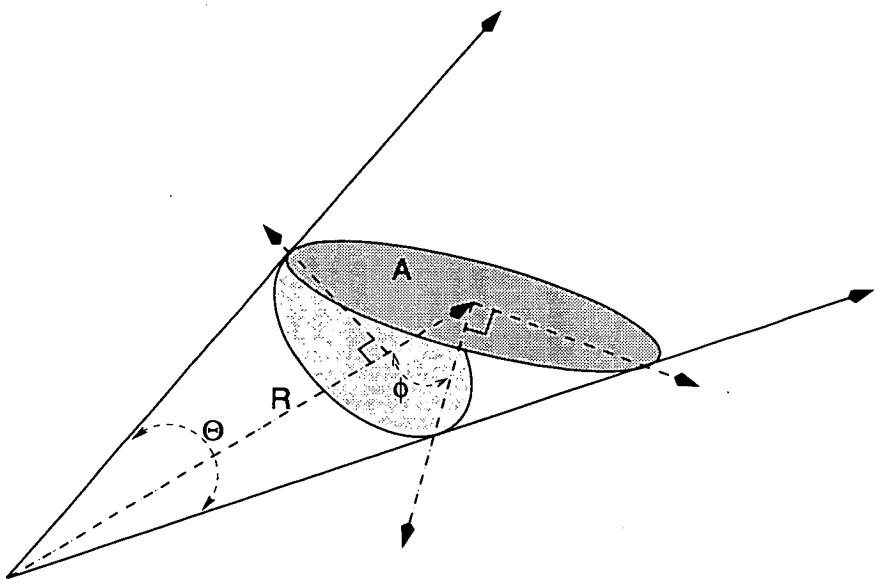


Figure 2.3: *The base area of a cone (lighter grey area) at a distance R from its origin is defined by its solid angle Θ. The general surface A can be reoriented towards the cone by the cosine of the angle, ϕ, the normal makes with the centre-projection of the cone.*

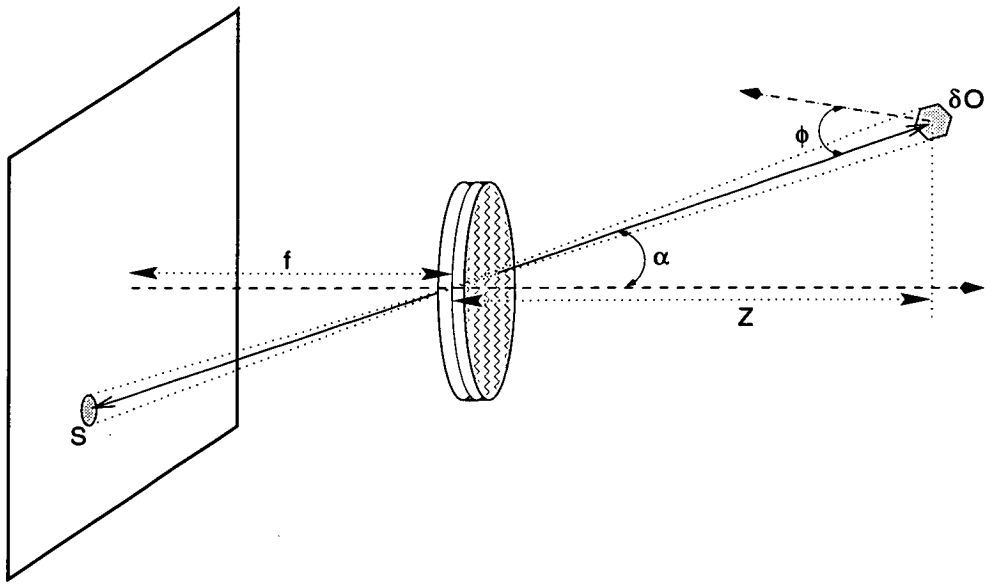


Figure 2.4: *The quantum of energy emerging from the scene patch, δO, is called irradiance and is concentrated on the sensor location at S. The measured quantum is called radiance.*

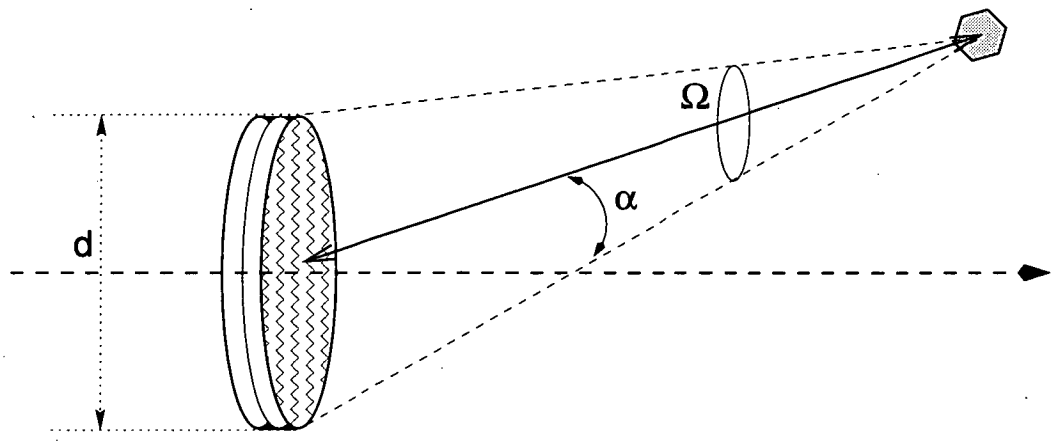


Figure 2.5: *The light catching potential of a lens is restricted by its aperture size d.*

example of a surface which concentrates the reflected light from a point source in only a very narrow angle. The mirror example clearly shows that the radiance of a surface depends on its physical properties, as well as the direction of illumination and the direction from which it is viewed. The surface physical property can be described by the Bidirectional Reflectance Distribution Function (BRDF) [101, 160]. In a coordinate system local to the scene point the BRDF can be defined in terms of the polar angle and azimuth of the incident and emitted rays,

$$f(\theta_i, \phi_i; \theta_e, \phi_e) = \frac{\delta L(\theta_e, \phi_e)}{\delta I(\theta_i, \phi_i)}, \tag{2.8}$$

where $\delta I(\theta_i, \phi_i)$ is the incident irradiance and $\delta L(\theta_e, \phi_e)$ is the emitted radiance.

The BRDF can be determined experimentally, but this is difficult and tedious because it has four degrees of freedom. Furthermore, it is impractical in realistic situations where the scene, and consequently the BRDF, is unavailable a priori. Notwithstanding this difficulty, the BRDF will lie between two extremes: one of all light being reflected at an angle determined by its incidence; and the other being light scattered independently of its angle of incidence.

An ideal specular surface reflects all light arriving from the direction (θ_i, ϕ_i) into the direction $(\theta_i, -\phi_i)$. This is typified by a perfect mirror surface, shown in Figure 2.6. In this case,

$$L(\theta_e, \phi_e) = I(\theta_i, -\phi_i), \tag{2.9}$$

the irradiance of the surface is equal to the radiance of the source and hence appears as a virtual image of the source. An ideal Lambertian surface is one that appears equally

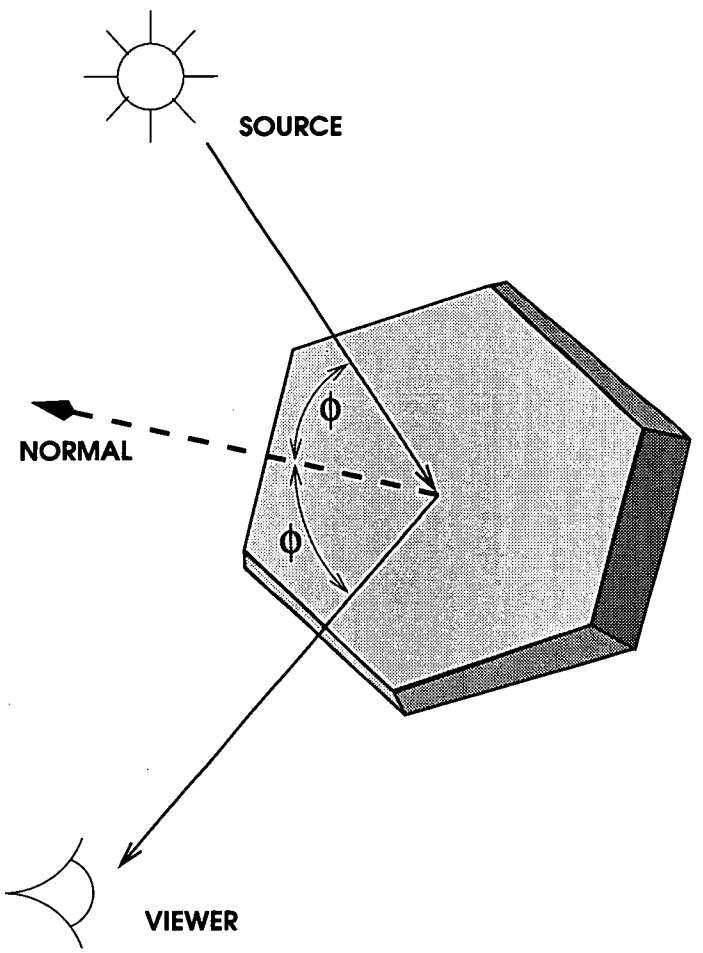
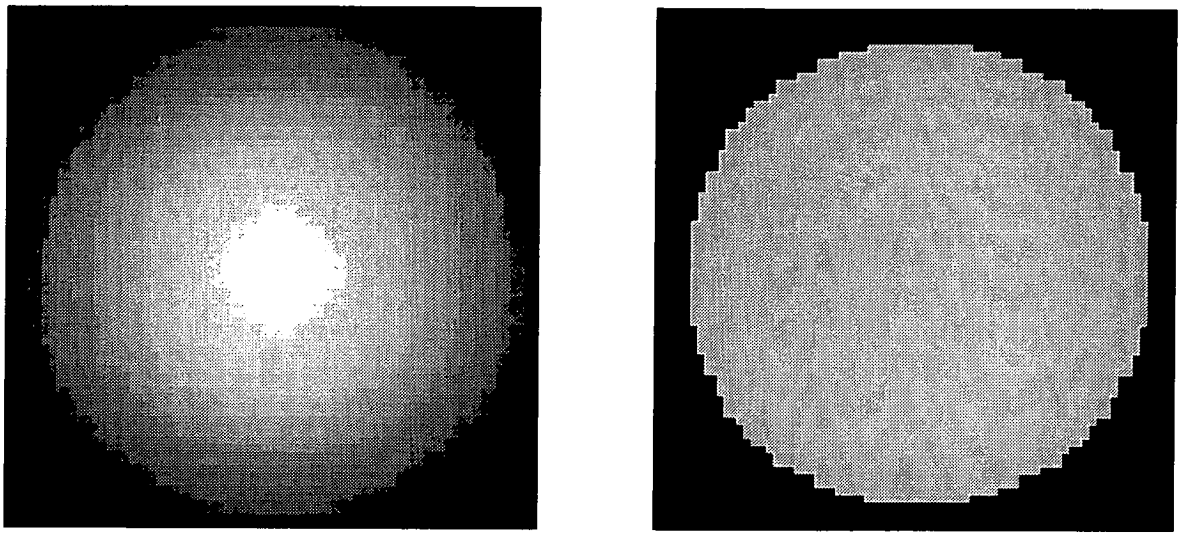


Figure 2.6: *Specular reflection ensures that the reflected light is concentrated in a direction of $-\phi$ from its angle of incidence.*



(a) (b)

Figure 2.7: *Lambertian reflective spheres: (a) illuminated by a point source, (b) by an extended or diffuse light source.*

bright from all viewing directions and reflects all incident light. In this case the BRDF is a constant. Horn [7] shows that for a Lambertian surface illuminated by a point source,

$$L = \frac{1}{\pi} I \cos\theta_i, \tag{2.10}$$

which is Lambert's cosine law of reflectance from matte surfaces. For an extended or diffuse light source,

$$L = \frac{1}{\pi} I_o. \tag{2.11}$$

Figure 2.7 shows two examples of a Lambertian sphere lit by a point source and an extended source. In the case of a point source the image surface appears curved due to the brightness variation, Figure 2.7(a). The image taken under an extended light source appears as a flat disk, Figure 2.7(b). In this case all sense of depth is lost in the image, a phenomena often referred to as "white out".

2.2 Camera Motion

In the introduction a number of image sequence examples, along with the associated needle diagrams, were shown. The needle diagram is an illustration of how different objects or parts within the view of the imaging sensor had moved over time. This movement can arise from either motion within the scene, movement of the sensor, or both. In the practical case

of weather forecasting for instance, a geostationary satellite (stationary sensor) monitors moving cloud patterns and ocean currents. However, in the example of an autonomous robot navigating though an automated storeroom the sensor is moving though a static scene. Most authors argue that the distinction is relative and dependent only on one's point of reference. However, as presented in the previous section on image formation, the BRDF of a surface is functionally dependent on the orientation and position of the scene to the light sources. Therefore the two forms of motion, from a radiometry point of view, are not equivalent. We consider this issue in greater detail in Section 2.4 where a number of common assumptions are considered. For consistency in this work the situation where the scene is chosen to remain static while the camera is allowed to move is the primary mode of operation.

Camera position and orientation in the three-dimensional world are described by 6 parameters. Position is most conveniently represented in rectangular coordinates, (X, Y, Z). Orientation is usually represented in terms of the angles, roll (α), tilt (β) and pan (γ) (also called roll, pitch and yaw), but may also be described as Euler parameters or quaternions. Roll, pitch, yaw angles usually lead to a rotation matrix description, but a quaternion representation affords a number of significant advantages, in particular a more compact description (see Appendix B). Quaternions are the form used in Chapter 5 when we discuss camera motion in greater detail.

If the camera is now allowed to move with constant translation component,

$$T = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{2.12}$$

and constant rotation component,

$$\Omega = \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix}, \tag{2.13}$$

then the general motion, or velocity of the camera can be written as,

$$P' = P \times \Omega - T, \tag{2.14}$$

where P is the position and P' is the instantaneous rate of change in position. The derivation of Eq.(2.14) is given in Chapter 5.

Egomotion, meaning self-motion, is often available in many practical situations to constrain the problem of measuring changes in the observed world. This issue is developed at length in Chapter 5. Firstly though, in the following two sections it is necessary to show the relationship between camera motion, the image plane irradiance pattern, and our ability to measure that change.

2.3 Optical Flow

The movement of the camera in the three-dimensional domain gives rise to a movement of the irradiance pattern falling on the two-dimensional image plane. This changing brightness pattern can be thought of as the "pixels" moving. The way in which the pattern changes can be determined from the perspective projective process that maps the three-dimensional world into the two-dimensional domain of sampled still frame pictures. Optical flow is defined to be the two-dimensional motion field resulting from the projective mapping of the three-dimensional motion field, Figure 2.8. The flow components at a point in the image plane are defined as,

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{-1}{Z}(fU + xW) - f\Omega_2 - y\Omega_3 + \frac{x}{f}(y\Omega_1 - x\Omega_2) \\ \frac{-1}{Z}(fV + yW) + f\Omega_1 + x\Omega_3 + \frac{y}{f}(y\Omega_1 - x\Omega_2) \end{bmatrix}. \tag{2.15}$$

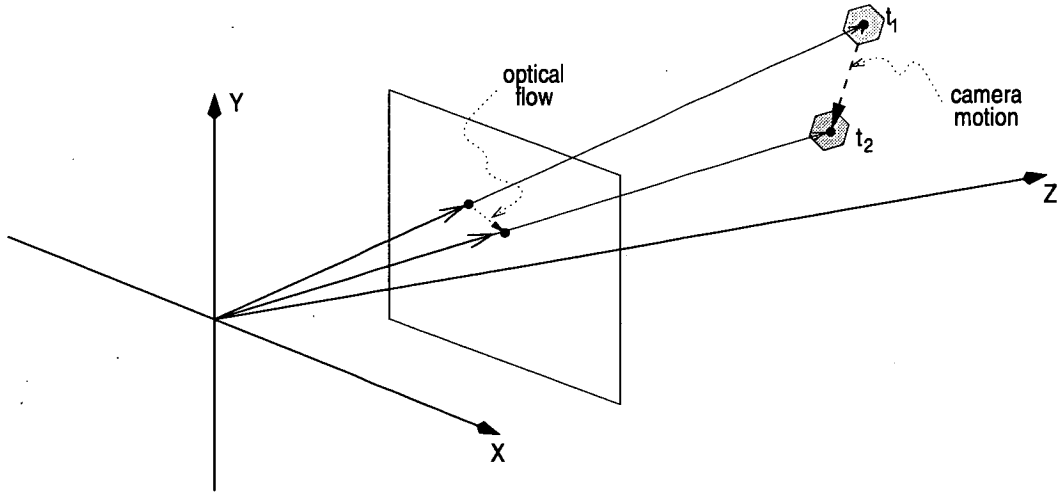


Figure 2.8: *The optical flow of a point is geometrically related to the camera motion by the structure of the scene.*

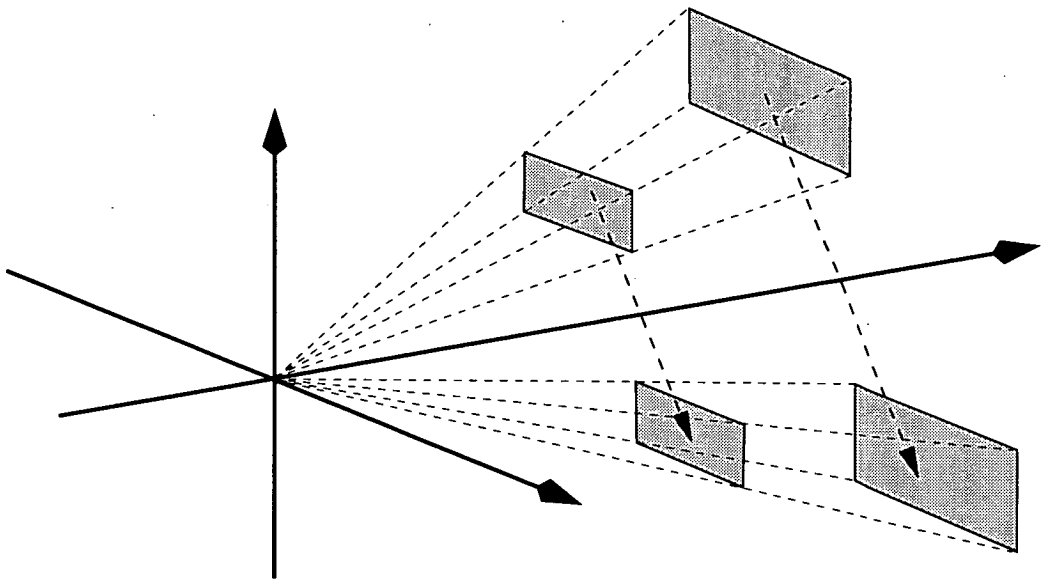


Figure 2.9: *Measurement of optical flow at a point is relative to the depth of the scene point. The relative velocity of a distant scene patch cannot be distinguished from that of a smaller object closer to the camera, despite the absolute difference between the velocities.*

It should be noted that the rotational motion on its own is insufficient to make depth judgments, since only the translational components of the motion are dependent upon the scene depth Z. The optical flow is therefore an inverse function of scene depth, provided there is some translational motion. This phenomena is apparent when driving a car where the distant hills approach slowly, while stationary objects nearby on the side of the ride appear as if they are moving much more quickly. Eq.(2.15) is also a non-unique mapping; as evidenced by the projection of a box being identical to the projection of one twice as big and twice as far from the sensor and moving at twice the speed. Hence, objects at large distances moving quickly cannot be distinguished from closer objects moving slowly, see Figure 2.9.

Science-fiction and action movie makers take full advantage of this ambiguity to add "apparent realism" to their special effects. By the same token the true depth can only be estimated to within a constant scale factor. Additional information about the physical size or velocity of a component of the scene, together with the relative depth is, however, sufficient to solve for the exact scene depth. As relative depth is estimated from the optical flow, accurate depth estimates depend on the availability of an accurate optical flow field. The following section outlines a number of assumptions and difficulties that must be overcome in order to estimate optical flow from a sequence of pixellated images.

2.4 Image Flow

In this thesis a distinction is drawn between the terms optical flow, which derives from the geometry of the scene and the sensor motion, and image flow. Image flow is the *measured* two-dimensional motion of the observed brightness pattern (image sequence) as it changes on the image plane and therefore only approximates the optical flow. Depending on their underlying assumptions different algorithms will produce different image flows.

Most current algorithms developed for accurately estimating the disparity between images rely on the following relationship holding [229],

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t). \tag{2.16}$$

$I(x, y, t)$ is the measured brightness of a scene point (X, Y, Z) when imaged by the camera at the point (x, y) and for the timed position t. Likewise, $I(x + \delta x, y + \delta y, t + \delta t)$ is the measured brightness of the same surface point (X, Y, Z) on the image plane for the camera at timed position $t + \delta t$. If the two images are captured at different camera positions, then $(\delta x, \delta y)$ is the measured two-dimensional disparity between the two images for the three-dimensional surface point (X, Y, Z).

In order for Eq.(2.16) to hold, the following assumptions based on image formation theory are required:

- Lambertian Reflectance Model

  The surface reflectance properties given by the BRDF, are Lambertian. For specular reflective surfaces the measured brightness varies with the viewing azimuth angle as shown in Eq.(2.9). As a result the requirement of Eq.(2.16) cannot be met. Lambertian reflective surfaces, on the other hand, ensure that equal energy is radiated in all directions of azimuth, see Eq.(2.10).

- Static Scene *(Point Source Illumination)*

  The three-dimensional scene must remain static with respect to the light source, only the camera is allowed to move. We see from Eq.(2.10) that the radiance of a Lambertian surface illuminated by a point source varies with the azimuth angle between source and surface. Figure 2.10 shows as example of a non-moving Lambertian sphere for which the optical flow is zero everywhere. However,

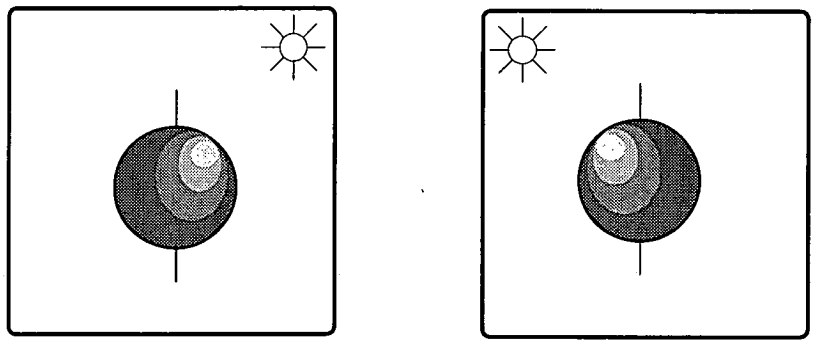


Figure 2.10: *A moving light point source gives rise to an apparent motion when the true optical flow is zero.*

because the point light source moves the brightness pattern on the sphere varies resulting in a non-zero image flow.

OR

- Diffuse Lighting *(Extended Source Illumination)*

  In order to overcome the restrictive requirement of a static scene, extended light source illumination is required. The measured brightness of a Lambertian reflective surface illuminated by an extended light source does not vary significantly with azimuth of the incident light rays or the azimuth of viewing, Eq.(2.11). Therefore the light reflected by a point on the scene towards the camera is unchanged by motion.

- Spatially Linear Camera Behaviour

  In order for uniform behaviour over the entire array of the image plane the lens system should not have any distortions or aberrations, and the photo-receptors that make up the image plane should be identical in performance. To some extent both of these errors can be removed by post-distortion to remove aberrations and non-uniform characteristics. Line-jitter and non-uniform read-out noise from CCD arrays are, however, much more difficult physical requirements to eliminate [60, 136].

In practice none of these requirements are met [171]. Nevertheless the fundamental assumption made by all image flow algorithms is that the change observed in the image intensity is due solely to motion, i.e. that Eq.(2.16) holds exactly.

2.5 The Aperture Problem

There is a much more fundamental difficulty, than those due to a failure to meet the underlying assumptions, that must be overcome to ensure that significant deviation of the image flow from the optical flow does not occur. This difficulty is known as the aperture problem and is used to describe situations where both components of the optical flow cannot be measured [7, 9, 15]. It arises because of a fundamental information loss due to the perspective transform, and because not all pixels contribute an equal amount towards the information content of the image *sequence*. As an extreme example consider the case of a spinning sphere with a Lambertian surface illuminated by either a stationary point or extended source, Figure 2.11. In this case the optical flow is non-zero. However, the sphere will "appear" to be stationary and thus has an image flow that is zero everywhere. In this example the spatial information content at the edges of the sphere is high, but the temporal information content is zero.

The information content of a pixel can be measured by the change of pixel value both spatially and temporally. To measure change it is necessary to consider a neighbourhood, sub-block, window, region or sub-"aperture" of the image. Windowed portions of different images are shown in Figure 2.12(a)–(d), each sub-figure illustrates a different aspect of the aperture problem.

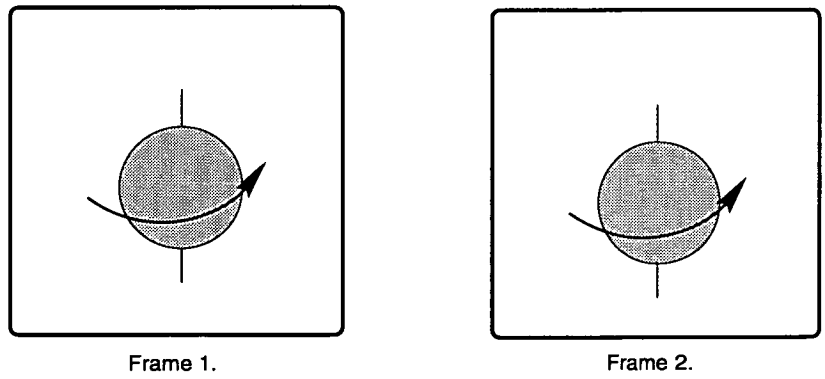


Figure 2.11: *A spinning Lambertian sphere illuminated by a diffuse light source appears stationary despite its true motion.*

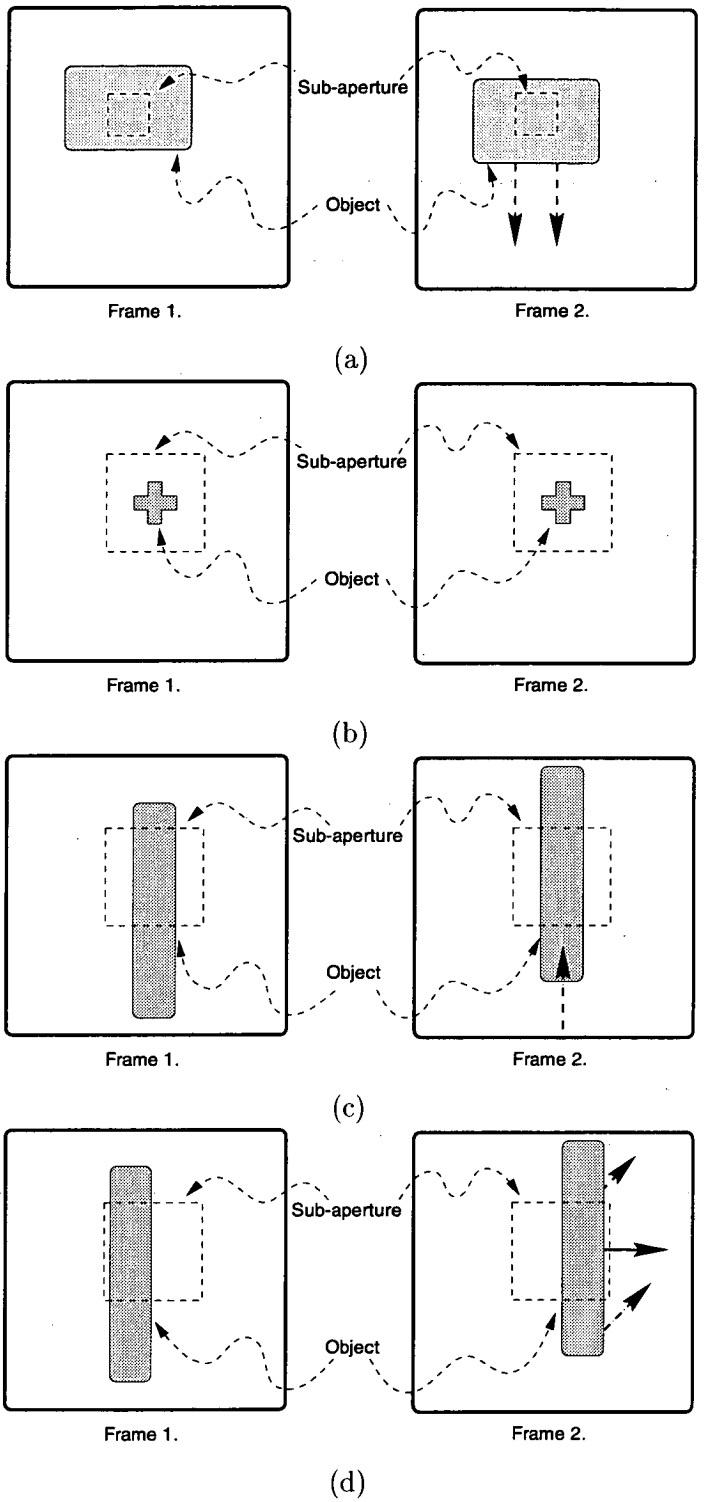


Figure 2.12: *Instances of the aperture problem: (a) a bland field of view, (b) a stationary object, (c) an "apparent" stationary object, (d) observing only the vernier flow.*

A major difficulty in computing image flow occurs in those regions of the image plane that are bland or lack detail, Figure 2.12(a). Consider the problem associated with observing the image plane surface through a finite aperture centred around the (x, y) pixel of a noisy sensor, shown as a dashed box. The Taylor series expansion of the image brightness pattern can be written as,

$$\begin{aligned} I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + I_x(x,y,t)\delta x + I_y(x,y,t)\delta y + I_t(x,y,t)\delta t \\ +I_{xx}(x,y,t)\frac{(\delta x)^2}{2} + I_{xy}(x,y,t)\delta x\delta y + I_{yy}(x,y,t)\frac{(\delta y)^2}{2} \\ +I_{xt}(x,y,t)\delta x\delta t + I_{yt}(x,y,t)\delta y\delta t \\ +I_{tt}(x,y,t)\frac{(\delta t)^2}{2} + ..., \end{aligned} \tag{2.17}$$

where subscripted variables denote partial differentiation with respect to the subscript computed at the point (x, y, t). If the aperture is very small, the surface is indistinguishable from a constant intensity equal to $I(x, y)$, simply because the effects of the higher order terms in the Taylor series are much smaller than the variations due to noise. In this situation no useful information can be obtained about the motion of the surface.

If we allow the aperture to increase in size, then the contributions of the first order terms of the Taylor series eventually become larger than the noise, whereupon an overall slope is discernible above the noise. In this case we can form a single equation,

$$I(x+\delta x, y+\delta y, t+\delta t) - I(x,y,t) = I_x(x,y,t)\delta x + I_y(x,y,t)\delta y + I_t(x,y,t)\delta t. \tag{2.18}$$

Provided the size of the window is large enough so that non-zero terms of the partial derivatives I_x and I_y exist, the motion term (u, v) can be estimated, where,

$$u = \frac{\delta x}{\delta t}, \tag{2.19a}$$

$$v = \frac{\delta y}{\delta t}. \tag{2.19b}$$

This is the first requirement in overcoming the aperture problem.

In Figure 2.12(b) we observe that a significant change in pixel brightness occurs spatially but not temporally. Again no motion is observed, however in this example the object is

indeed stationary. Consequently, it could be concluded that only rapid spatial change in illuminance of the image region is required for reliable detection of image flow. The previous drawn conclusion is too imprecise, as can be demonstrated in the example shown in Figure 2.12(c). In this example the window applied to the two frames appears unchanged despite the real motion that has occurred. Furthermore, the subimage within the window is neither bland nor lacking in detail. Hence, the conclusion made above must be qualified by the statement:

> "Motion can only be determined in the direction parallel to the spatial gradient of the illuminance, since orthogonally to the gradient there is little or no variation in illuminance."

This problem of only being able to determine the component of the velocity vector parallel to the gradient is another instance of the aperture problem. As a result image flow estimates often only approximate the vernier component of the optical flow, i.e. the component parallel to the optical gradient, see Figure 2.12(d).

When dealing with an aperture of this size there is only one equation to solve for two unknowns. The only solution to this difficulty is to further enlarge the size of the aperture in some way. If this is done, we can for example use changes in the spatial gradients to determine the additional conservation of gradient intensity equations,

$$\frac{d}{dt}(I_x) = I_{xx}u + I_{xy}v + I_{xt} = 0, \tag{2.20a}$$

and

$$\frac{d}{dt}(I_y) = I_{xy}u + I_{yy}v + I_{yt} = 0. \tag{2.20b}$$

Thus, it is evident that provided the size of the aperture is large enough that non-zero second order spatial gradients terms exist, then both velocity components can be estimated. This is the second requirement in overcoming the aperture problem.

By way of example consider the case of a camera moving parallel to its optical axis towards a box. The optical flow from the geometry of the problem is a diverging field, as illustrated in Figure 2.13. However, the observed and estimated image flow does not radiate uniformly from the centre of the image plane. Only at points of high curvature, such as corner points,

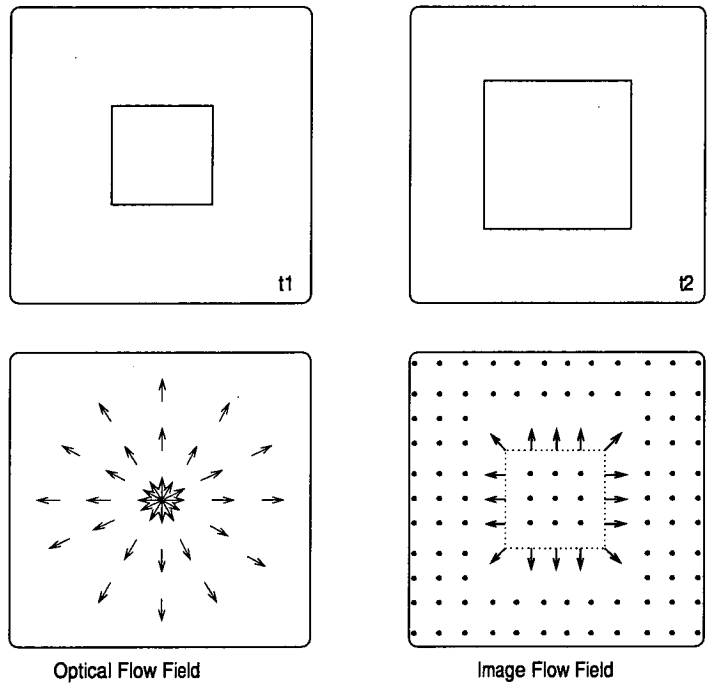


Figure 2.13: *Translating along the optical axis gives rise to a diverging optical flow field. However, the image flow field may differ significantly from the optical flow field and only at points of high curvature, such as corner points, are more reliable estimates made.*

is a more accurate estimate possible. Image flow measurement techniques are therefore not only required to compute the vernier flow field, but also attempt to overcome limitations imposed by the aperture problem. The only way of achieving this is to use larger aperture sizes.

The aperture size can be further increased to improve velocity estimates, and also to allow the measurement of other geometric qualities of the optical flow field, such as the kinematics of the motion field [8]. However, as the size of the aperture is increased spatial resolution of the flow field is inevitably reduced. This is an example of the uncertainty principle; *both* location and velocity cannot be estimated with infinite precision [21, 29]. Rather than fit more complicated models over a larger aperture, the approach taken here is to use the smallest aperture capable of estimating u and v. More complex behaviour can then be recovered from the resulting higher resolution image flow field.

2.6 Sampling Theorem for Moving Pictures

The image capture process involves discretizing the continuous intensity function of the radiating scene in time and space, after it has been convolved with the point spread function of the lens system [17,21,29]. The discrete spatial units are formed from a regularly spaced array of finite sized elements referred to as pixels. Ideally, however, we take the continuous three-dimensional intensity function and multiply it with a three-dimensional impulse train to form the sampled image sequence. The purpose of the following discussion is to extend the well established one-dimensional sampling theory to three-dimensions.

Consider a one-dimensional signal $f(t)$ and the impulse sampling train,

$$\delta_\infty = \sum_{n=-\infty}^{\infty} \delta(t + nT), \tag{2.21}$$

then the sampled signal is,

$$\begin{aligned} f_s(t) &= f(t) \sum_{n=-\infty}^{\infty} \delta(t + nT) \\ &= \sum_{n=-\infty}^{\infty} f(nT)\delta(t + nT), \end{aligned} \tag{2.22}$$

where T is the sampling period. Now, consider the affect of sampling on the Fourier spectrum of the original signal $f(t)$. Denoting the Fourier transform pair as,

$$f(t) \leftrightarrow F(\omega), \tag{2.23}$$

where the Fourier spectrum $F(\omega)$ is band-limited to ω_m as shown in Figure 2.14(a). The Fourier spectrum of the sampled signal is given by the transform pair,

$$f_s(t) \leftrightarrow F(\omega) \odot \mathcal{F}\left\{ \sum_{n=-\infty}^{\infty} \delta(t + nT) \right\}. \tag{2.24a}$$

The Fourier transform of the impulse sampling train is given by,

$$\mathcal{F}\left\{ \sum_{n=-\infty}^{\infty} \delta(t + nT) \right\} = \frac{\omega_0}{2\pi} \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0), \tag{2.24b}$$

where $\omega_0 = \frac{2\pi}{T}$ is the sampling frequency. Hence the sampled spectrum is,

$$F_s(\omega) = \frac{2\pi}{T} F(\omega) \odot \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0), \tag{2.24c}$$

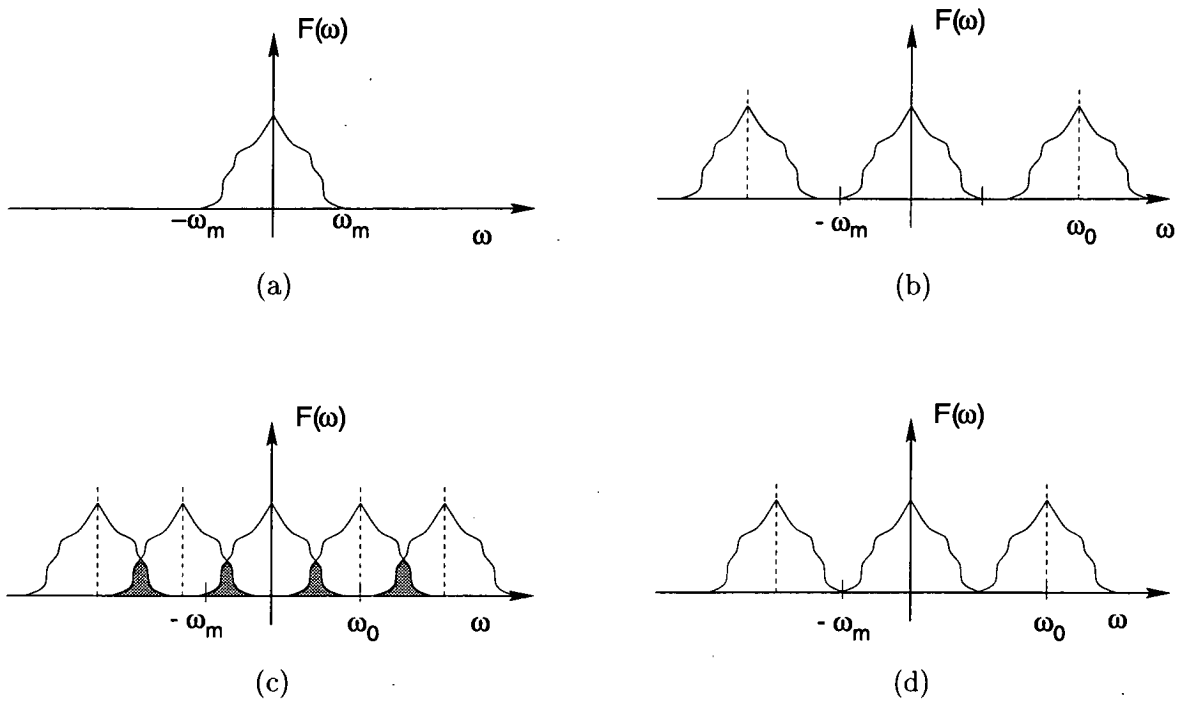


Figure 2.14: *Effect of the sampling process: (a) The original continuous signal spectrum, (b) the spectrum with sampling $\omega_0 > 2\omega_m$, (c) aliasing spectrum $\omega_0 < 2\omega_m$, (d) critical sampling $\omega_0 = 2\omega_m$.*

and as shown in Figure 2.14(b) it consists of the periodic extension of the spectrum $F(\omega)$ where $\omega_0 > 2\omega_m$. Clearly low-pass filtering to remove the periodic extensions yields the original unsampled spectrum and permits recovery of $f(t)$. Figure 2.14(c) shows the case for $\omega_0 < 2\omega_m$, the periodic parts of the spectrum now overlap and the original signal is now corrupted and cannot be retrieved by low-pass filtering, this is called aliasing. Figure 2.14(d) illustrates critical sampling, where $\omega_0 = 2\omega_m$. Recovery of the original signal is possible only with an ideal low-pass filter with a cutoff frequency of ω_m.

The minimum sampling frequency required to prevent aliasing must just exceed twice the highest frequency component in the spectrum. This minimum sampling frequency is called the Nyquist frequency or rate ω_N. Low-pass filtering of the continuous signal $f(t)$ is required before sampling to ensure the signal is bandlimited and complies with the Nyquist criteria,

$$\omega_N \geq 2\omega_m. \tag{2.25}$$

In practical systems sampling must be done with a finite length train of rectangular pulses

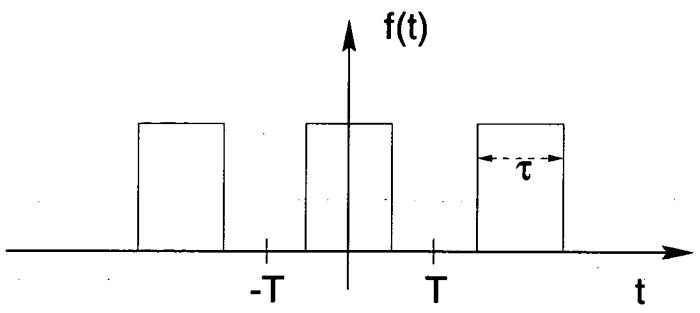


Figure 2.15: *Natural or practical sampling is achieved with pulses of finite width (τ) over a fixed interval of time T.*

of finite width τ, and not an infinitely long train of impulse functions, see Figure 2.15. The finite length of the rectangular pulse train reflects that in practice sensors are finite in size. This in turn also implies that the signal is bandlimited. The Fourier series of the rectangular pulse train is,

$$S(t) = \sum_{k=-K}^{K} c_k e^{j2\omega k}, \tag{2.26a}$$

$$c_k = \frac{\tau}{T} \frac{\sin(k\pi\tau/T)}{k\pi\tau/T}. \tag{2.26b}$$

Hence the Fourier transform pair for natural sampling,

$$f(t)S(t) \leftrightarrow \sum_{k=-K}^{K} c_k F(\omega - k\omega_0). \tag{2.26c}$$

Therefore, the only difference between natural sampling and impulse sampling is; in natural sampling there are a finite number of periodic extensions, where each side-lobe of $F(\omega)$ is weighted by the diminishing Fourier series coefficients, c_k, of the rectangular pulse. Therefore if the original signal is bandlimited, and the Nyquist criteria is satisfied, aliasing will not occur.

In the multi-dimensional case the Nyquist criteria must be satisfied in each of the signal's dimensions. So in the case of a two-dimensional (x, y) signal or image, the Nyquist spatial frequency pair (k_{x_N}, k_{y_N}) must meet the criteria:

$$k_{x_N} \geq 2k_{x_m} \qquad \text{and} \qquad k_{y_N} \geq 2k_{y_m}, \tag{2.27}$$

where (k_{x_m}, k_{y_m}) are the two maximum spatial frequency components in directions x and y respectively. For a three-dimensional (x, y, t) image sequence where ω denotes the temporal

frequency component the criteria is,

$$k_{x_N} \geq 2k_{x_m}, \qquad k_{y_N} \geq 2k_{y_m} \quad \text{and,} \qquad \omega_N \geq 2\omega_m. \tag{2.28}$$

In practice the optics of a vision system bandlimit the spatial frequencies of the image, although rarely, if ever, are CCD arrays Nyquist sampled [7, 107]. This leads to mild aliasing which can often be ameliorated by post-sampling low-pass filtering or smoothing.

The fundamental assumption, on which all image flow algorithms are based, is that changes of the image intensity function in time are solely due to motion. This assumption leads to a relationship between the spatial and temporal sampling rates as a function of the motion. Fourier analysis considers the signal to be a sum of orthogonal sinusoidal components, consider one such component,

$$s(x, y, t) = \cos(k_x x + k_y y + \omega t). \tag{2.29a}$$

If the change of the image over time is due solely to motion then the sinusoid can be rewritten as,

$$s(x, y, t) = \cos(k_x(x + ut) + k_y(y + vt)), \tag{2.29b}$$

and therefore,

$$\omega = k_x u + k_y v. \tag{2.29c}$$

The maximum temporal frequency is thus related to the maximum spatial frequencies by,

$$\begin{aligned}
& \omega_m = k_{x_m} u + k_{y_m} v \leq \tfrac{1}{2}\omega_N \\
\Longrightarrow \quad & \tfrac{1}{2}k_{x_N} u + \tfrac{1}{2}k_{y_N} v \leq \tfrac{1}{2}\omega_N \\
\Longrightarrow \quad & \sqrt{k_{x_N}^2 + k_{y_N}^2}\sqrt{u^2 + v^2}\cos\theta \leq \omega_N \\
\Leftrightarrow \quad & \sqrt{u^2 + v^2}\cos\theta \leq \frac{\omega_N}{\sqrt{k_{x_N}^2 + k_{y_N}^2}},
\end{aligned} \tag{2.30}$$

where θ is the angle between the velocity vector and the spatial sampling vector. Equation (2.30) is referred to as the Nyquist velocity equation. In the best case, for a spatial sampling of 1 per pixel in both the x and y directions and a temporal sampling of 1 per frame, the maximum permissible speed is 1 pixel per frame [79]. Picture elements that move faster than this may be aliased and consequently incorrectly estimated by image flow algorithms. This is, however, a particularly stringent restriction on admissible velocities and impractical

in many applications. Techniques for effectively increasing the Nyquist velocity are required and this can be achieved in three different ways:

- Increasing the temporal sampling rate
- Decreasing the spatial sampling rate
- Increasing temporal sampling and decreasing spatial sampling

In practice image sequences are often captured at a fixed temporal rate leaving variation of the spatial sampling rate as the sole mechanism for changing the Nyquist velocity. A higher Nyquist velocity requires a reduction in the number of spatial samples, with a consequential loss in spatial resolution of the image and estimated flow field. Fortunately, through the use of multi-resolution techniques, it may be possible to obtain velocity estimates greater than the Nyquist limit for a particular resolution, a topic briefly discussed in Chapter 7.

2.7 Numerical Differentiation

As discussed in the previous section on the aperture problem, spatial and temporal gradients of the image brightness pattern are an implicit requirement in explaining the relationship between the image sequence and the image flow. Moreover, a large proportion of algorithms that make image flow estimations do so based on gradient measurements computed from the raw image data. Consequently, in this and the following section, a closer look is taken at the practical issues involved in computing image gradients.

In practice exact derivatives must be approximated by some numerical method, such as interpolation functions or finite difference approximation to the derivatives [8]. In the former approach the image intensity surface is fitted by interpolation functions for which exact derivatives exist. Classes of interpolation functions that fit this requirement include polynomial functions, and splines [96, 210, 211]. The fitting process involves determining the coefficients of the interpolation function and consequently the coefficients of the partial derivatives of the interpolation function. Therefore any errors in fitting the interpolation functions to the image surface will result in errors in the derivative estimates.

In the latter case, numerical differentiation by finite differences is the process of estimating a function $g(x) = (f(x + \Delta x) - f(x))/\Delta x$ for a small but finite Δx, and is the non-limiting

case of exact differentiation. Three options for computing finite differences exist: forward, backward and central differences. In their original work Horn and Schunck[102] used forward differences as the means of approximation. This method has the disadvantage of approximating the derivative at the mid-points between samples. Similarly backward differences also approximate the derivative at the mid-points between samples. Central differences, on the otherhand, approximate the derivative at sample points. This thesis presents results based on the use of central finite differences, due to their ease of understanding, speed in computing and popularity with other authors. By choosing this path direct comparisons with other works are possible, as well as with the optical flow velocities which are usually only available at pixel locations.

If we consider the Taylor series expansion about the point x of the function $f(x+\Delta x)$,

$$f(x+\Delta x) = f(x) + \Delta x f'(x) + \frac{(\Delta x)^2}{2!} f''(x) + \frac{(\Delta x)^3}{3!} f'''(x) + \frac{(\Delta x)^4}{4!} f^{iv}(x) + \frac{(\Delta x)^5}{5!} f^{v}(x) + ..., \tag{2.31}$$

then a three-point central difference evaluates the function $f(x)$ at x, $x-\Delta x$ and $x+\Delta x$,

$$\begin{aligned} c_{-1}f(x-\Delta x) + c_0 f(x) + c_1 f(x+\Delta x) &= (c_{-1}+c_0+c_1)f(x) + (c_1 - c_{-1})\Delta x f'(x) \\ &\quad + \tfrac{1}{2}(c_{-1}+c_1)(\Delta x)^2 f''(x) \\ &\quad + \tfrac{1}{6}(c_1 - c_{-1})(\Delta x)^3 f'''(x) + ... \end{aligned} \tag{2.32a}$$

To obtain the first order derivative approximation, one divides through by Δx and selects values for c_{-1}, c_0 and c_1 so that the right-hand side of Eq.(2.32a) approximates $f'(x)$. To solve for the three unknowns, three equations are needed, which for optimality are,

$$\begin{aligned} 0 &= c_1 + c_0 + c_{-1} \\ 1 &= c_1 - c_{-1} \\ 0 &= c_1 + c_{-1}. \end{aligned} \tag{2.32b}$$

The solution is $c_0 = 0$, $c_{-1} = -c_1$ and $|c_1| = \frac{1}{2}$. Hence the three-point difference mask is,

-1/2	0	1/2

The error of this approximation is associated with the higher order derivatives,

$$\mathcal{O}_1(\Delta x, 3) = \tfrac{1}{6}(\Delta x)^2 f'''(x) + \tfrac{1}{120}(\Delta x)^4 f^v(x) + ... \tag{2.33}$$

Using the same process, the five-point first order difference can be specified. This yields the optimal solution for the five-point difference mask of,

1/12	-2/3	0	2/3	-1/12

with approximation error due to higher order derivatives of,

$$\mathcal{O}_1(\Delta x, 5) = (\tfrac{1}{60}\cdot\tfrac{2}{3} + \tfrac{32}{60}\cdot\tfrac{-1}{12})(\Delta x)^4 f^v(x) + (\tfrac{1}{2520}\cdot\tfrac{2}{3} + \tfrac{128}{2520}\cdot\tfrac{-1}{12})(\Delta x)^6 f^{vii}(x) + ... \tag{2.34}$$

Note that since the kernel shows an odd symmetry, $c_{-i} = -c_i$, the problem can be reduced to finding only $n = (\kappa - 1)/2$ coefficients. In general, the c_n coefficients of a κ-tap first order central difference filter, can be found by solving the following system of equations [8],

$$2\begin{bmatrix} 1 & 2 & 3 & ... & n \\ 1 & 8 & 27 & ... & n^3 \\ 1 & 32 & 243 & ... & n^5 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 2^{2n-1} & 3^{2n-1} & ... & n^{2n-1} \end{bmatrix}\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{2.35}$$

The error in the optimal κ-point difference can be approximated by,

$$\mathcal{O}_1(x, \kappa) = \Upsilon(\kappa)(\Delta x)^{\kappa-1}\frac{\partial^\kappa f(x)}{\partial x^\kappa} \tag{2.36a}$$

where,

$$\Upsilon(\kappa) = \frac{2}{\kappa!}\sum_{n=1}^{\frac{\kappa-1}{2}}(c_n n^\kappa). \tag{2.36b}$$

The error measure of Eq.(2.36a) makes the assumption that,

$$(\Delta x)^{\kappa-1}\frac{\partial^\kappa f(x)}{\partial x^\kappa} \gg \sum_{n=1}^{\infty}(\Delta x)^{\kappa-1+2n}\frac{\partial^{\kappa+2n} f(x)}{\partial x^{\kappa+2n}}, \tag{2.36c}$$

and hence the higher order derivative terms can be omitted.

The solutions for the seven and nine point central differences are:

-1/60	9/60	-45/60	0	45/60	-9/60	1/60

$$\mathcal{O}_1(\Delta x, 7) = \frac{1}{140}(\Delta x)^6 f^{vii}(x) \tag{2.37}$$

3/840	-32/840	168/840	-672/840	0	672/840	-168/840	32/840	-3/840

$$\mathcal{O}_1(\Delta x, 9) = \frac{-1}{630}(\Delta x)^8 f^{ix}(x). \tag{2.38}$$

The optimal second order derivative masks can be formed by cascading (convolving) two optimal first order difference filters together. For instance, the $f''(x)$ derivative formed from the 3-point single difference mask $[-1/2, 0, 1/2]$ is given by the 5-point mask,

1/4	0	-1/2	0	1/4

The error associated with this approximation is given by,

$$\mathcal{O}_2(x, \kappa) = \frac{2}{(\kappa+1)!}(\Delta x)^{\kappa-1}\frac{\partial^{\kappa+1} f(x)}{\partial x^{\kappa+1}}\sum_{n=1}^{\frac{\kappa-1}{2}}\left(c_n n^{\kappa+1}\right), \tag{2.39}$$

based on a similar assumption to that given by Eq.(2.36c).

2.8 Regularization and Noise

In any practical system where $f(x)$ is subject to noise, the difference $f(x+\Delta x) - f(x)$ will also be corrupted by the additive noise. If this difference term is divided by the small step size Δx, then the noise in $f(x+\Delta x) - f(x)$ is amplified and the derivative may become numerically unstable. This is a well known problem which requires regularization, a process

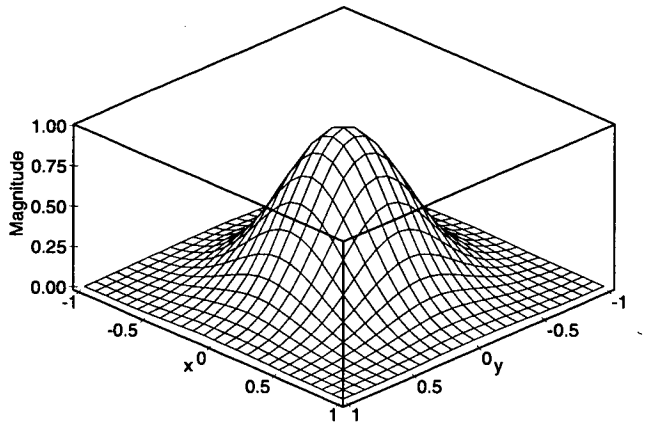


Figure 2.16: *Two-dimensional Gaussian smoothing function.*

usually performed by smoothing the function or surface before a derivative is taken. This is done by a process of convolution [5, 17, 21],

$$\begin{aligned} I^s(x,y,t) &= I(x,y,t) \odot S(x,y,t) \\ &= \iiint I(x',y',t')S(x-x',y-y',t-t')dx'dy'dt', \end{aligned} \tag{2.40}$$

where $I(x,y,t)$ is the sensor image at time t and $S(x,y,t)$ is a smoothing function. Conventionally Gaussian smoothing is employed (see Figure 2.16) [5, 8, 199],

$$S(x,y,t) = \frac{1}{(2\pi)^{3/2}\alpha^2\beta} e^{-\frac{x^2+y^2}{2(\alpha^s)^2} - \frac{t^2}{2(\alpha^t)^2}}, \tag{2.41}$$

Increasing the parameter α^s extends the level of smoothing equally in both directions of the spatial domain, with α^t controlling the level of temporal domain smoothing. Independent spatial and temporal smoothing provides a compromise between spatial and temporal resolution for a fixed overall smoothing requirement. By putting a greater requirement on spatial smoothing less frames are required, due to the smaller temporal smoothing requirement. Consequently, a slower frame capture rate can be used for a fixed temporal resolution, or a higher temporal resolution for a fixed capture rate. This requirement, however, provides a reduced spatial resolution. Conversely, reducing the spatial smoothing improves spatial resolution and decreases temporal resolution. In all cases the smoothing function is

normalized so that,

$$\iiint S(x,y,t)dx\,dy\,dt = 1. \tag{2.42}$$

The derivatives are also usually formed by a process of convolution with a derivative kernel, as discussed in the previous section, with the estimate of the x-derivative ultimately formed by,

$$\begin{aligned} I_x(x,y,t) &= I(x,y,t) \odot S(x,y,t) \odot D_x(x,y,t) \\ &= I(x,y,t) \odot D_x^s(x,y,t). \end{aligned} \tag{2.43}$$

Smoothing has the dual effects of suppressing noise components and also ensuring the differentiability of discontinuities. The first effect can easily be computed in the case where $I(x,y,t)$ is subject to independent noise on each pixel with a variance equal to $(\sigma_n)^2$. In this case the noise on the x-derivative is equal to,

$$\begin{aligned} (\sigma_n)_x^2 &= (\sigma_n)^2 \int_{-\infty}^{\infty}\iiint \left(D_x^s(x,y,t)\right)^2 dx\,dy\,dt \\ &= (\sigma_n)^2 \int_{-\infty}^{\infty}\iiint \frac{x^2\left(e^{-\frac{x^2+y^2}{2(\alpha^s)^2}-\frac{t^2}{2(\alpha^t)}}\right)^2}{8\pi^3(\alpha^s)^8(\alpha^t)^2} dx\,dy\,dt \\ &= \frac{\sigma_n^2}{16\pi^{3/2}(\alpha^s)^4(\alpha^t)}, \end{aligned} \tag{2.44}$$

for a Gaussian smoothing function. The noise of the higher-order derivatives of x are shown in Table 2.1. The y-derivative are easily computed using a similar technique. The significant reduction in the effect of the noise on the derivatives as α^s and α^t increases is readily apparent. Although in practice both the image and convolutional kernels are finite and discrete, the results do not differ significantly from Table 2.1 provided α^s and α^t are greater than 1.

The second effect of regularization is to ensure that the function is differentiable. This also has the effect of reducing the magnitude of the derivatives, as shown in Figure 2.17 which illustrates smoothing on a step discontinuity in both the x and y directions. Regularization gives rise to a smooth differentiable "greyscale corner", Figure 2.17(b), but both slope and curvature decrease as the smoothing increases further, an effect apparent in Figure 2.17(c) and (d).

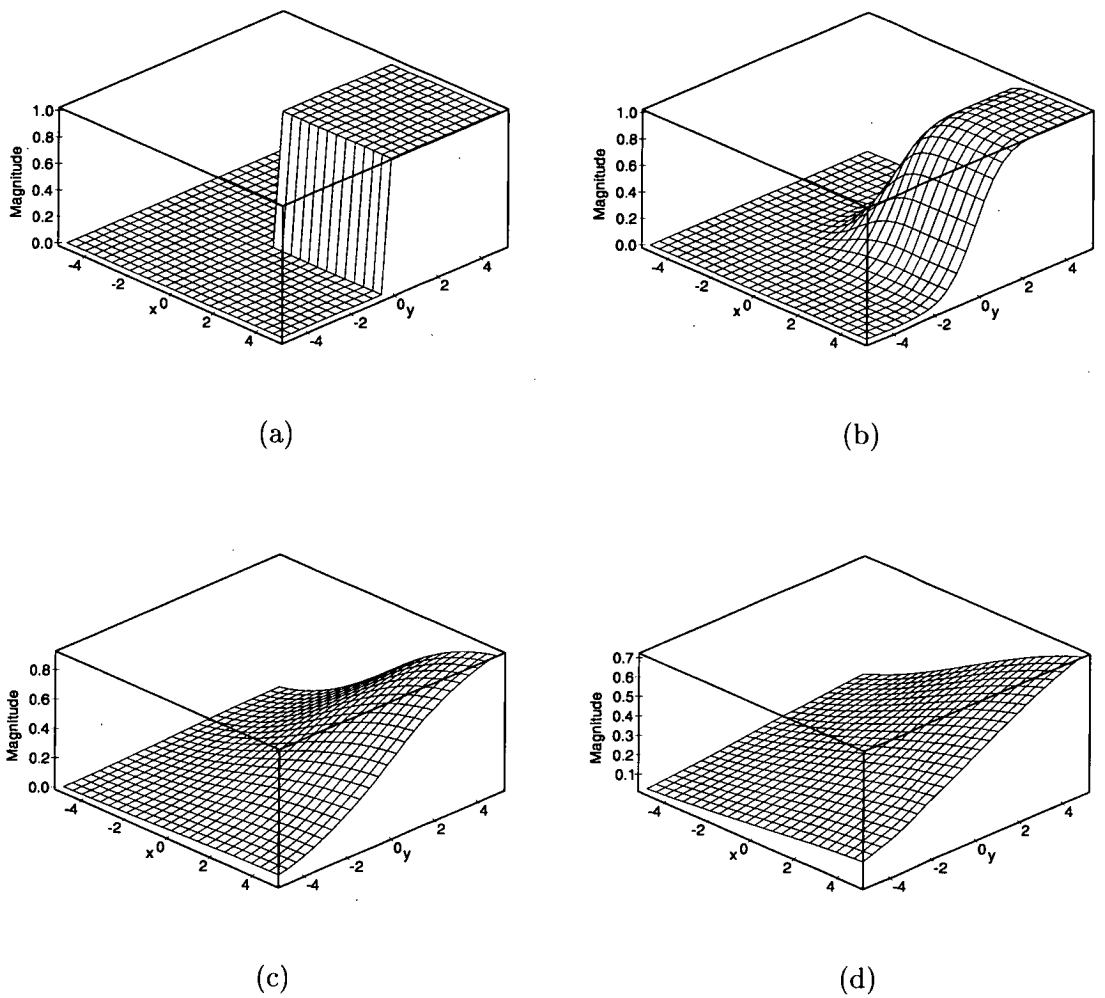


Figure 2.17: *Regularization of a greyscale corner using Gaussian smoothing. (a) Input discontinuity, (b) $\alpha = 1$, (c) $\alpha = 3$, (d) $\alpha = 5$.*

While it is not possible to quantify the decrease in the derivatives for an arbitrary real sequence, it can be done for a translating random field sequence with variance $(\sigma_r)^2$. For ease we also require $\alpha = \alpha^s = \alpha^t$. In this case the three-dimensional data set and its derivatives are constant in the direction defined by the vector (u, v, t). We now choose a Cartesian coordinate system (x', y', t') where the t'-axis is parallel to the vector (u, v, t). Smoothing now results in the signal power of the x-derivative being,

$$(\sigma_r)_x^2 = (\sigma_r)^2 \iiint_{-\infty}^{\infty} \left(D_x^s(x', y', t')\right)^2 dx'dy'dt'$$

$$= (\sigma_r)^2 \iint_{-\infty}^{\infty} \left(D_x^s(x', y')\right)^2 dx'dy', \tag{2.45}$$

since the signal is constant in the t' direction. This results in the variance of the derivatives, or signal power, decreasing with smoothing as shown in Table 2.2. Should there also be independent random noise present on each pixel in the image, then the signal to noise ratio of the derivatives increases proportionally to α. The accuracy of the velocity estimates would thus be expected to improve as the smoothing is increased.

Excessive smoothing, however, does introduce other difficulties. Firstly, an implicit assumption is being made that the velocity is constant over the area being smoothed. This is not the case with real data and must inevitably introduce errors into the velocity estimates. Secondly, the derivatives are smaller and subject to the effects of both the loss of numerical precision and the finite approximation of the derivative kernel. As the smoothing increases, these effects eventually dominate the gains made by reducing the effects of sensor noise and thus the image flow estimate no longer improves with increased smoothing. The net-effect is ultimately a decreasing SNR due to increasing "numerical" noise.

Derivative	Noise amplification
$D_x^s(x, y, t)$	$\frac{1}{16\pi^{3/2}\alpha^4\beta}$
$D_{xx}^s(x, y, t)$	$\frac{3}{32\pi^{3/2}\alpha^6\beta}$
$D_{xy}^s(x, y, t)$	$\frac{1}{32\pi^{3/2}\alpha^6\beta}$
$D_{xt}^s(x, y, t)$	$\frac{1}{32\pi^{3/2}\alpha^4\beta^3}$

Table 2.1: *The effect of noise on Gaussian regularized derivatives. The parameters α and β control the level of spatiotemporal smoothing.*

Derivative	Signal amplification
$D_x^s(x,y,t)$	$\frac{1}{8\pi\alpha^4}$
$D_{xx}^s(x,y,t)$	$\frac{3}{16\pi\alpha^6}$
$D_{xy}^s(x,y,t)$	$\frac{1}{16\pi\alpha^6}$

Table 2.2: *The magnitude of Gaussian regularized derivatives for a translating random field sequence.*

Smoothing also has other practical implications, these include: the removal of mild aliasing and a reduction of the image flow resolution. The available image flow is lost in two ways as the size of the smoothing kernel is increased. Firstly at the boundaries of the image, where by convention a border of $\kappa/2$ is required for a κ length mask. Secondly, because of the blurring effect of smoothing, which tends to spread and average the pixel intensity values and consequently the computed image flow. Thus, for reasons of positional accuracy (resolution) it is desirable to keep smoothing to a minimum, a factor in conflict with the need to estimate derivatives accurately.

2.9 Error Measures

It is vitally important to have a consistent and logical method for comparing different image flow algorithms. For quantitative comparisons this requires a well behaved metric, one which handles extreme values equally as well as values that are not extreme. Almost universally adopted by researchers is a mean square difference between the expected and estimated flow fields. Other alternative error measures include the RMS error and mean absolute difference. In more recent years quantitative comparisons of the estimated flow field to the optical flow have been made using Fleet and Jepson's mean angular velocity error metric [4, 88]. An example includes a recent summary of the existing state of the art of image flow techniques in the work by Barron et al.[56].

Fleet projected the optical flow vector at a point into a three-dimensional space-time coordinate system as $a = (u, v, 1)$. The corresponding three-dimensional image flow is defined

as $\hat{a} = (\hat{u}, \hat{v}, 1)$. The angular separation between these two vectors is given by,

$$\cos \phi^e = a \cdot \hat{a}$$
$$\phi^e = \cos^{-1}\left(\frac{a \cdot \hat{a}}{|a|\,|\hat{a}|}\right) \tag{2.46}$$

There are, however, a number of problems with this error metric. Symmetric deviations of the estimated image flow from the expected optical flow yields an asymmetric angular error [167]. Consider two image flow estimates of $(0.5, 0.5)$ and $(1.0, 0.5)$ for an expected optical flow of $(0.75, 0.5)$. Both estimates have the same absolute difference of 0.25 pixel/frame, however the angular error of the first estimate is 1.07°, while the second estimate has an angular error of 0.87°. Additionally, the measure is non-linear and consequently a change in scale of the magnitude of the image flow does not result in a linear change in scale of the error metric. The angular error for a scaling of the image flow by k is given by,

$$\phi^e(k) = \cos^{-1}\left(\frac{k(a \cdot \hat{a} - 1) + 1}{\sqrt{k^2(|a| - 1) + 1}|\hat{a}|}\right).$$

Hence quite large magnitude errors may not result in a significant change of error metric to that of lesser-errors. Both Fleet[4] and Barron et al.[54] acknowledge this "bias" behaviour, suggesting however, that the metric handles extreme behaviour (large and very small speeds) in a more appropriate manner than measures of vector difference.

The major problem with mean error metrics, including the mean angular error, is that they are unduly sensitive to gross errors and can therefore be dominated by a few relatively high errors emphasizing these measures. Indeed this result is evident in the following chapters. For this reason a median square error measure is also used in this thesis,

$$\text{Median}\left\{|a - \hat{a}|^2\right\} = \text{Median}\left\{(u - \hat{u})^2 + (v - \hat{v})^2\right\}. \tag{2.47}$$

In general all error measures have their own limitations and drawbacks. Hence, for completeness and to allow comparisons with Barron et al. [54–56] results, error estimates are provided using: mean and median differences, as well as Fleet and Jepson's mean and median angular error measures.

Throughout this entire work image flow and corresponding error measures are evaluated at every pixel location within the image, except within the border region of the images. This corresponds to an image flow density of 100%. By convention a border of $\kappa/2$ is required

for any image convolved with a mask of length κ. The size of the border, therefore, is determined by the length of any preprocessing filters applied to the image, for example smoothing filters and gradient kernels. Unless otherwise stated, kernel sizes, apertures and borders are measured in pixels.

Chapter 3

Review of Image Flow Techniques

In the preceding chapter image flow was defined as an estimate of optical flow derived from image intensity measurements. This chapter reviews the various types of image flow algorithms and shows that in essence the five methods commonly referred to in the literature can be further reduced to two methods: correlation or gradient based. Either method may be implemented in the space-time or Fourier Domains. Some authors have also proposed hybrid methods, based on a two step process using both correlation and gradient methods [166, 209].

3.1 Feature Matching

Feature matching algorithms for computing image flow depend on matching tokens (distinctive features such as edges) between successive frames [154, 195, 204, 228]. Due to the aperture problem these schemes usually only utilize intensity corner points, i.e. points of high curvature, as valid tokens since these provide velocity information in more than one direction. The ends of lines, points of intersection and isolated points are all good examples of intensity corner points [72, 147, 206, 225, 227]. Feature matching schemes are required to identify and track each token in a robust manner in order to compute the image flow (Figure 3.1).

Difficulties arise when many tokens are identified since a multitude of different matches between consecutive frames become possible. In order to overcome this correspondence

problem additional constraints are required to reduce the complexity of possible matches. A partial simplification is offered when the two views are separated by a known distance and direction, as is the case of binocular or stereo vision approaches.

Figure 1.1 illustrates a binocular vision system. The baseline, which joins the centre points of the two camera lenses, is particularly important for constraining the point correspondences in two images. In particular, corresponding points will lie on a line parallel to the projection of the baseline onto the image plane. This is known as the epipolar line. This approach offers a much simplified problem, as it reduces the task of identifying correspondences to matches along a line as opposed to an area. It is still possible for more than one valid match to occur along the line. A complete and unique solution in this case is only possible by using three or more cameras that are not arranged collinearly, see Figure 3.2. Furthermore, if the cameras are aligned such that the epipolar lines are parallel to the rows and columns of the sensors, computationally efficient line searches can be implemented.

With monocular vision systems where the camera motion is known, an epipolar line constraint can also be implemented. In practice, however, the direction of motion is only known to within some degree of error, as a result potential correspondence points lie within a region parallel to the epipolar line. In many cases where the camera motion is unknown a priori the epipolar line constraint cannot be used. In order to uniquely match features for this case additional complex information must be encoded with each token, in order to distinguish it from all the others. Murray and Buxton suggest the use of edgels; elements of an edge occupying one pixel that encode distinguishing characteristics such as intensity, orientation, activity, and curvature [11, 193]. This process has a number of significant draw backs:

- The process of finding and encoding targets is complex.
- Computationally intensive.
- Feature points in an image are usually sparsely distributed.

These difficulties are significant since the more complex a task becomes the more difficult its implementation in a general and robust way. Furthermore, sparsely distributed points result in a sparsely computed velocity field, which for many applications is unacceptable. For instance, a sparse flow field when used in motion encoding for HDTV provides poor

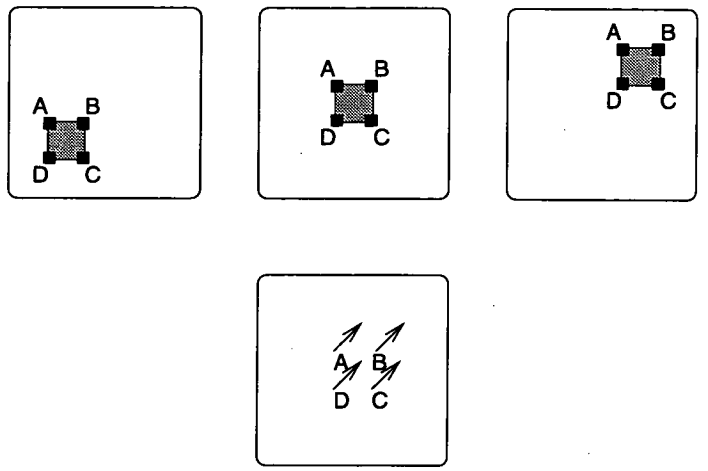


Figure 3.1: *Point tracking requires identifying common points in the image sequence and matching these from frame to frame.*

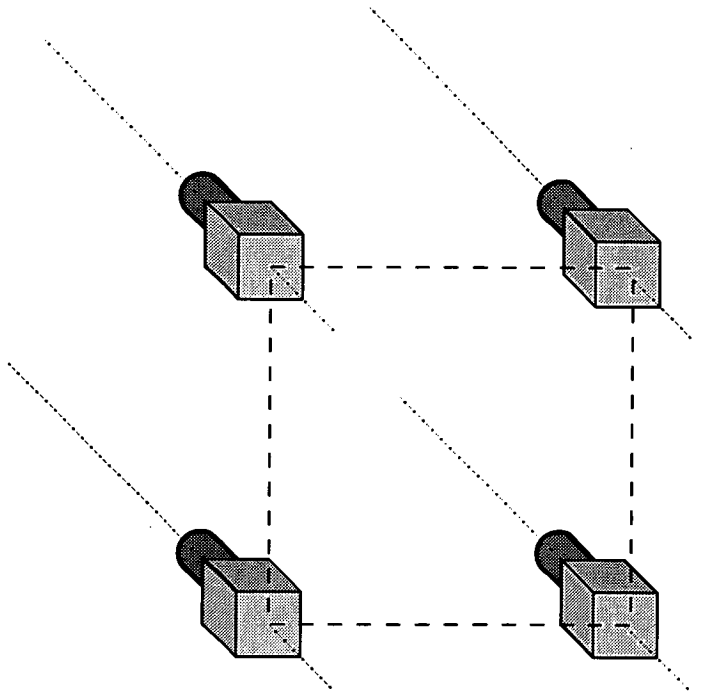


Figure 3.2: *A quad vision system as used by Jarvis[115] offers the ability of overcoming the correspondence problem.*

levels of motion compression. In a robotics problem, sparsity results in an inability to navigate reliably. Consequently, alternative techniques which produce a dense estimate throughout the image must be considered.

3.2 Correlation Matching

In the feature matching process for arbitrary camera motion, it proves necessary to encode not only a token's location but also its characteristics. Comparison of tokens then becomes a process of measuring the similarity of the token to others in a regions. This in essence is a correlation process. From our understanding of the aperture problem more than a single point is required to solve for the image flow, since curvature at a point, for instance, must be computed from a neighbourhood of pixels. Textures are often as good a feature as edges and corner points, and such structures cannot be identified and coded at the pixel level. Instead an image patch or window is required, with the correspondence problem being solved by matching patches from one frame to the next. Such methods construct a window about each pixel in the image, which is then compared to similar windows in the successive frame. These windows not only contain texture information, but also the edge and corner information previously encoded by the feature extraction process. This approach therefore simplifies the feature extraction stage to one of forming correlation windows, thereby eliminating the need to identify and label distinctive image features.

Figure 3.3 shows a typical situation with a correlation window in Frame #1 and a second in Frame #2. One can proceed to find the translation of a patch by either using a maximum correlation or a minimum mean-square difference, [11,13,119,135,183,186]. The response

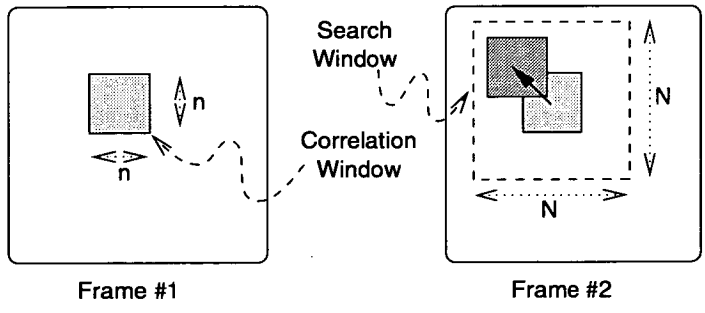


Figure 3.3: *The correlation approach: windows of size $n \times n$ are compared from Frame 1 and Frame 2. The window may be offset within a possible range of $N \times N$.*

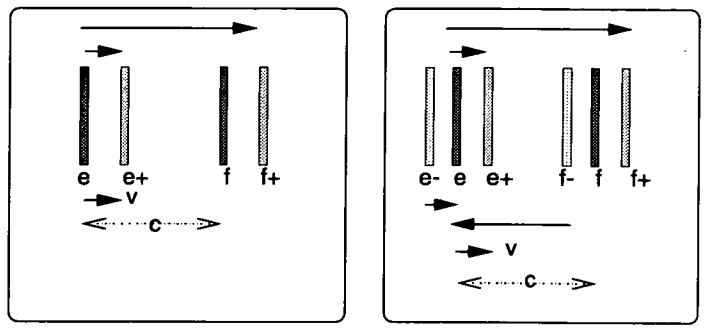


Figure 3.4: *Autocorrelation problem: (a) the forward two frame problem, (b) is eliminated by the central three frame correlation.*

peak in the correlation signal thereby corresponds to the shift between image segments and hence gives the image flow. The maximum correlation process estimates the shift by finding the maximum value in,

$$R(u,v) = \sum_{i=-n}^{n} \sum_{j=-n}^{n} (I(x+i, y+i, t_1) I(x+u+i, y+v+j, t_0)) . \tag{3.1}$$

The approach taken by Scott[183], Murray and Buxton[11] and later by Singh and Allen[186] is to use the mean-square difference. An error indicating the matching between the two correlation frames is computed using the sum-of-squared differences,

$$E(u,v) = \sum_{i=-n}^{n} \sum_{j=-n}^{n} (I(x+i, y+j, t_1) - I(x+u+i, y+v+j, t_0))^2 \tag{3.2}$$

where the correlation displacement (u,v) at the point (x,y) is constrained by $-\frac{N-1}{2} \leq u,v \leq +\frac{N-1}{2}$. This is repeated for all possible displacements in the search window and assembled into a $N \times N$ error distribution.

There is, however, a problem with using only two frames in the correlation process, known as the autocorrelation problem [11]. Consider two dark solid edges, e and f with a separation of c, as shown in Figure 3.4(a). These edges move with a velocity of v to the right giving the lighter lines, e^+ and f^+ respectively. Two matches are possible for the edge e either to e^+ or f^+. If averaged this results in an incorrect estimate of the image flow of $v + c/2$.

Murray and Buxton[11] suggests using a central three frame correlation to overcome the

two frame autocorrelation problem. In this case the matching error is given by,

$$E(u,v) = E^{1,0}(u,v) + E^{-1,0}(-u,-v), \tag{3.3a}$$

where,

$$E^{1,0}(u,v) = \sum_{i=-n}^{n}\sum_{j=-n}^{n} \left(I(x+i,y+j,t_1) - I(x+u+i,y+v+j,t_0)\right)^2 \tag{3.3b}$$

$$E^{-1,0}(-u,-v) = \sum_{i=-n}^{n}\sum_{j=-n}^{n} \left(I(x+i,y+j,t_{-1}) - I(x-u-i,y-v-j,t_0)\right)^2 . \tag{3.3c}$$

This matching is symmetrical and has the advantage that the computed image flow corresponds directly to the central frame $I(x,y,t_0)$. Two further advantages of this approach are also afforded. Firstly, a degree of noise suppression is provided since more data is employed in forming the image flow estimates. Secondly, autocorrelation problems which may occur between two frame matchings are eliminated. In Figure 3.4(b) the edges from the backward frame, $I(x,y,t_{-1})$, are shown as the light lines e^- and f^- respectively. Two matches are possible with the backward frame giving an average velocity of $v-c/2$, which when averaged with the forward frame estimate via Eq.(3.3a) gives the correct image flow of v.

The minimum point in the error distribution corresponds to the offset shift where the best match is achieved. When using the raw data provided by the correlation process this limits the accuracy in offset and image flow to the nearest pixel. Interpolation is required to achieve sub-pixel accuracy. In order to enhance the peak location for the interpolation process, the error distribution is passed through an adaptive non-linear filter [186],

$$R(u,v) = e^{-\kappa E(u,v)}, \tag{3.4}$$

designed to sharpen peak response points. The parameter κ controls the degree of non-linearity observed. The peak is then found by maximizing the response distribution $R(u,v)$ rather than minimizing $E(u,v)$.

In their original work Singh and Allen[186] used,

$$\kappa = -\log(0.95)/\min\{E(u,v)\}, \tag{3.5}$$

so that the peak of $R(u,v)$ corresponds closely to 1, thus eliminating potential numerical difficulties as the error approaches zero. The choice of a response distribution was motivated

primarily on computational reasons; it is well behaved as the error approaches zero, it varies continuously between one and zero as well as providing some sharpening of the peak. Clearly as long as the response distribution is unimodal, and does not lie close to the edge of the search window, accurate location of the peak is possible. Moreover, Singh and Allen state that they found their results to be largely insensitive to the arbitrary 0.95 parameter.

However, their method is deficient when the error values approach zero, since Eq.(3.5) is undefined for an error of zero. Given the discrete nature of the data this is not unlikely. In order to overcome this Barron et al. implemented a stabilizing condition,

$$\kappa = 0.0085 \qquad \text{if } \min\{E(u,v)\} < 1 \times 10^{-6}, \tag{3.6}$$

and also suggested the use of the smallest non-zero error term. An alternative and more stable definition of κ would be,

$$\kappa = \begin{cases} \dfrac{-\log(\eta)}{\max\{E(u,v)\}} & \text{if } \max\{E(u,l)\} \neq 0 \\ 0 & \text{otherwise} \end{cases}, \tag{3.7}$$

where η is a small value, such as 10^{-6}. Stability is insured because the response distribution is bounded between 0 and 1, with the additional computational benefit that $\max\{E\}$ is less likely to be close to zero than $\min\{E\}$. Furthermore, high error values are scaled to be close to zero.

The response distribution can be interpreted statistically as a likelihood distribution. Each point in a search area is a valid candidate for the true velocity (displacement) and therefore a point with a small response is less likely to be a match than a point with a high response. Thus, the response distribution can be converted to a probability distribution in (u,v) space,

$$P(u,v) = \frac{R(u,v)}{\sum_{i=-N}^{N}\sum_{j=-N}^{N} R(i,j)}, \tag{3.8a}$$

the image flow is then computed by,

$$u = \sum_{i=-N}^{N}\sum_{j=-N}^{N} P(i,j)i \tag{3.8b}$$

$$v = \sum_{i=-N}^{N}\sum_{j=-N}^{N} P(i,j)j. \tag{3.8c}$$

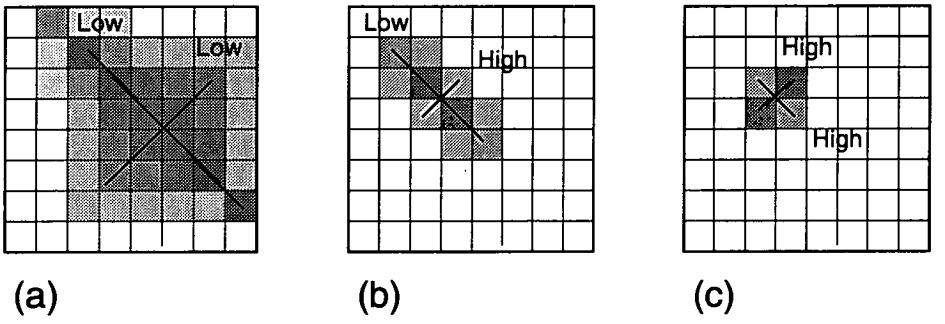


Figure 3.5: *Representative examples of the correlation error distribution over the search window; lighter the pixel the higher the error. The labels high and low refer to the confidence measures associated with the image flow velocity estimate, (a) in a largely uniform area, (b) near an edge, (c) near a corner.*

This corresponds to a mean rather than modal estimate of the true displacement. Similarly, the covariance of the image flow measure is given by the 2×2 matrix,

$$\xi(x,y) = \sum_{i=-N}^{N} \sum_{j=-N}^{N} P(i,j) \begin{bmatrix} (i-u)^2 & (i-u)(j-v) \\ (i-u)(j-v) & (j-v)^2 \end{bmatrix}. \tag{3.9}$$

Spectral decomposition of the inverse covariance matrix at each point then yields two confidence measures with associated eigenvectors. The orthogonal eigenvectors point in the directions of greatest and least confidence, with the former corresponding approximately to the direction parallel to the spatial gradient, Figure 3.5.

Singh and Allen added two sophisticated features to their algorithm to improve its performance: multi-resolution analysis and smoothing. Smoothing of the image flow is primarily used for propagating velocity information from areas where it is well known to less well known areas, although it also provides a noise reduction feature as well. Multi-resolution analysis is used to estimate large velocities which exceed the size of the correlation search window, or fail to meet the Nyquist sampling rate. A smoothed and subsampled version of the image sequence is produced, resulting in a lower-resolution image sequence which displays coarser features. The velocity estimates made at this lower scale can then be projected up to the higher scale. The subsampling process of the image sequence can be performed a number of times to form a pyramid of scaled images. This results in a coarse to fine adjustment process in the estimation of the flow field velocity. In their work Singh and Allen[186] produce a scale of images using an efficient Laplacian pyramid construct [67]. Their choice was motivated by the strong relationship between accurate flow field estimation and edge information in the image sequence. The Laplacian edge operator results in a pyramid of

edge detected images from which they make their image flow estimates. However, Barron et al.[54] noted in their work that the Laplacian pyramid constructs band-pass channels that contain substantial aliasing.

3.3 Spatial-Temporal Gradient Model

It is convenient to analyze the information contained from observing a small finite area of the image $I(x+\delta x, y+\delta y, t+\delta t)$, located around the point (x, y). This is most conveniently done by considering the Taylor series expansion of the image intensity around this point, Eq.(2.17). The essential relationship for the two frames $I(x, y, t)$ and $I(x, y, t+\delta t)$, given by Eq.(2.16), can be expressed as a differential equation of the light intensity,

$$\begin{aligned}
&\lim_{\delta t \to 0} \frac{I(x+\delta x, y+\delta y, t+\delta t) - I(x, y, t)}{\delta t} \\
&= \lim_{\delta t \to 0} \left(\frac{I(x, y, t+\delta t) - I(x, y, t)}{\delta t} + I_x(x, y, t+\delta t)\frac{\delta x}{\delta t} + I_y(x, y, t+\delta t)\frac{\delta y}{\delta t} \right) \\
&= 0, \qquad (3.10)
\end{aligned}$$

where the Taylor expansion of $I(x, y, t+\delta t)$ is taken only to first order based on the assumption that only the first order terms are significant. In the limit an under-determined constraint equation for the image flow is provided,

$$\frac{dI}{dt} = I_t + I_x u + I_y v = 0, \qquad (3.11)$$

where (u, v) is the image flow at the point (x, y, t).

Clearly there is a distinction between the terms dx/dt and $\delta x/\delta t$, the former is the instantaneous rate of change while the later is the average rate of change over a finite time. Nagel[158] goes to some length to distinguish between the two terms referring to them as image flow and displacement rate respectively. In practice, instantaneous quantities cannot be computed and for this reason image flow is defined for the displacement rate or average rate of change. Normalizing the frame rate to $\delta t = 1$ allows direct comparison between image flow measurements made by gradient and correlation based algorithms.

Eq.(3.11) can be rewritten in terms of the magnitude of the image flow and the angle, θ,

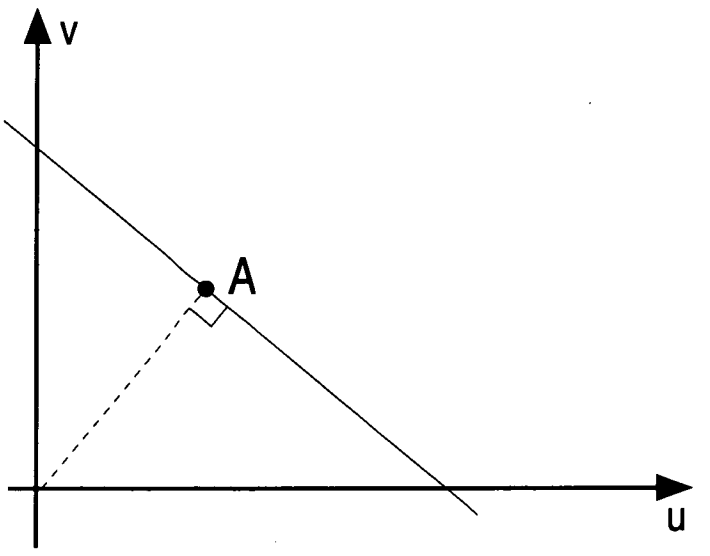


Figure 3.6: *The conservation of intensity, Eq.(3.11) constrains the velocity to the solid line on the $u-v$ plane. Horn and Schunck's initial velocity estimate is a point on the dash line segment between the origin and passing through A.*

that it makes with the spatial gradient,

$$|(u,\ v)| = \frac{-I_t}{\sqrt{I_x^2 + I_y^2 \cos\theta}}. \tag{3.12}$$

This mathematically illustrates that only the component of the image flow in the direction of the illuminance gradient, i.e. where $\cos(\theta) = 1$, can be estimated. This is referred to as the vernier or normal flow component. Image flow estimates that are equally consistent lie on a constraint line within the (u, v) space, as shown in Figure 3.6. Point A corresponds to the vernier flow component and is the minimum Euclidean distance from the origin. Therefore, in order to solve Eq.(3.11) fully a further constraint is required. It is the choice of additional constraints that differentiate between various spatial-temporal gradient or differential methods for estimating image flow.

The first differential solution to the aperture problem was the approach of Horn and Schunck[102] who obtained the image flow field by minimizing the error function,

$$E^t = \iint \left(E^i + \gamma^2 E^{s_H}\right) dx\, dy, \tag{3.13a}$$

where,

$$E^i = (I_x u + I_y v + I_t)^2 , \tag{3.13b}$$

$$E^{sH} = \left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right), \tag{3.13c}$$

and γ is a Lagrange multiplier weighting constant that controls the relative importance of the error in the intensity constraint to that of the departure from smoothness. E^{sH} is a smoothness constraint and it is sufficient to ensure a unique solution on the constraint line shown in Figure 3.6. As the limits of integration are over the entire image Eq.(3.13c) is often referred to as a global smoothness constraint.

The smoothness constraint equation of Eq.(3.13c) can be rewritten in an alternative form that provides both additional insight into its purpose and a more computationally efficient form,

$$\iint (u_x^2 + u_y^2 + v_x^2 + v_y^2) dx\, dy \equiv (\bar{u} - u)^2 + (\bar{v} - v)^2, \tag{3.14}$$

where $\bar{u}, \bar{v}$ denote the average velocity components in the x and y directions respectively. Minimizing the error metric, E^t, with respect to the image flow leads to the equations,

$$\frac{\partial E^t}{\partial u} = 2\left(I_x u + I_y v + I_t\right) I_x - 2\gamma^2 \left(\bar{u} - u\right) = 0, \tag{3.15a}$$

$$\frac{\partial E^t}{\partial v} = 2\left(I_x u + I_y v + I_t\right) I_y - 2\gamma^2 \left(\bar{v} - v\right) = 0, \tag{3.15b}$$

which are solved simultaneously for the image flow components u and v,

$$u = \bar{u} - \frac{I_x \bar{u} + I_y \bar{v} + I_t}{I_x^2 + I_y^2 + \gamma^2} I_x, \tag{3.16a}$$

$$v = \bar{v} - \frac{I_x \bar{u} + I_y \bar{v} + I_t}{I_x^2 + I_y^2 + \gamma^2} I_y. \tag{3.16b}$$

We see from Eq.(3.16) that the image flow estimate at a point is dependent on the average image flow of the neighbourhood about that point. The average image flow in turn is dependent on the point estimates, thus leading to a "chicken-and-egg" problem. This can be resolved by employing a Gauss-Seidel iterative form to find (u, v), [7, 198]. Denoting $(\bar{u}^\tau, \bar{v}^\tau)$ to be the average velocity at the point (x, y) at iteration τ, then the computed flow at iteration $\tau + 1$ is,

$$\begin{bmatrix} u^{\tau+1} \\ v^{\tau+1} \end{bmatrix} = \begin{bmatrix} \bar{u}^\tau \\ \bar{v}^\tau \end{bmatrix} - \left(\frac{I_x \bar{u}^\tau + I_y \bar{v}^\tau + I_t}{I_x^2 + I_y^2 + \gamma^2} \right) \begin{bmatrix} I_x \\ I_y \end{bmatrix}. \tag{3.17}$$

Appropriate starting conditions for the iterations are $(\bar{u}^0, \bar{v}^0) = (0,0)$, which gives a first estimate of,

$$\begin{bmatrix} u^1 \\ v^1 \end{bmatrix} = -\left(\frac{I_t}{I_x^2 + I_y^2 + \gamma^2}\right)\begin{bmatrix} I_x \\ I_y \end{bmatrix}. \tag{3.18}$$

For the case of $\gamma = 0$,

$$\begin{bmatrix} u \\ v \end{bmatrix} = -\frac{I_t}{\sqrt{I_x^2 + I_y^2}}\frac{1}{\sqrt{I_x^2 + I_y^2}}\begin{bmatrix} I_x \\ I_y \end{bmatrix}, \tag{3.19}$$

which is the component of image flow in the direction of the intensity spatial gradients with magnitude that puts (u,v) on the constraint line at point A, see Eq.(3.12) and Figure 3.6. In the case of $\gamma \neq 0$, (u,v) also lies in the direction of the spatial gradient, shown by the dashed line segment between the origin and passing through A in Figure 3.6. Subsequent iterations may lead to estimates which do not lie on either line shown in Figure 3.6.

3.4 Fourier Methods

Both the correlation and differential techniques have also been implemented in the Fourier domain. Their Fourier equivalences are often referred to as the Fourier Energy and Fourier Phase methods respectively, [4,88,89,99,111,118,185,187]. Our understanding of the aperture problem shows us that the best image flow estimates are made at points in the image that are high in detail or structure. This information is embodied in the phase of the Fourier signal. The proponents of Fourier Phase methods argue, therefore, that as the Fourier phase function encodes only the structure of the image intensity they are largely insensitive to variations in total illuminance. Consequently, Fourier methods are much more stable to noise on the brightness measurements and should therefore provide more reliable image flow estimates.

The Fourier transform of a two-dimensional image signal $I(x,y,t)$ can be written as,

$$\mathcal{F}^{(2)}\{I(x,y,t)\} = I^0 e^{-j\phi(x,y,t)} \tag{3.20}$$

where I^0 is the magnitude and $\phi(x,y,t)$ the Fourier phase of the signal. The correlation of two images A and B is given by their cross-spectrum in the Fourier domain [17],

$$R = \mathcal{F}^{(2)*}\{I^B\}\mathcal{F}^{(2)}\{I^A\}. \tag{3.21}$$

The Fourier phase of images A and B are written as,

$$\phi^A = k_x x + k_y y, \tag{3.22a}$$

$$\begin{aligned}\phi^B &= k_x(x - \delta x) + k_y(y - \delta y) \\ &= k_x(x - u\delta t) + k_y(y - v\delta t) \\ &= k_x x + k_y y - \omega\delta t,\end{aligned} \tag{3.22b}$$

where δt is the time interval between frames, (k_x, k_y) the spatial frequency components and ω is the temporal frequency component. An underlying assumption is made that the image flow Fourier phase is linear,

$$\omega = k_x u + k_y v. \tag{3.23}$$

This is a simplifying assumption which is realistic provided the time difference between sampling of frames is small, it is also consistent with the space-time methods which assume linear variation of the image flow at a point. The cross-spectrum of the two signals can then be written as,

$$\begin{aligned}R &= |\mathcal{F}^{(2)}\{I^A\}|^2 e^{-j\omega\delta t}, \\ &= \left|\int_{-\infty}^{\infty} \mathcal{F}^{(3)}\{I\}d\omega\right|^2 e^{-j\omega\delta t},\end{aligned} \tag{3.24}$$

which relates the two-dimensional Fourier transform of the image to its three-dimensional equivalent for the planar wave phase function given in Eq.(3.22b). The integration performs an averaging of the temporal frequency which is recovered from R by,

$$\bar{\omega} = \tan^{-1}\frac{\mathcal{I}\{R\}}{\mathcal{R}\{R\}}. \tag{3.25}$$

Substitution into Eq.(3.23) provides the Fourier energy estimate of the image flow which is consistent with the vernier component of optical flow field at each point,

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{\bar{\omega}}{k_x^2 + k_y^2}\begin{bmatrix} k_x \\ k_y \end{bmatrix}. \tag{3.26}$$

In essence the problem being solved is that given by Eq.(3.23) which bears a close resemblance to that given by Eq.(3.11). In particular it should be noted from Eq.(3.22b) that

the following relationships hold:

$$\frac{\partial \phi}{\partial x} = k_x, \tag{3.27a}$$

$$\frac{\partial \phi}{\partial y} = k_y, \tag{3.27b}$$

$$\frac{\partial \phi}{\partial t} = \omega. \tag{3.27c}$$

This result demonstrates the equivalence of correlation and differential approaches to estimating image flow. In the Fourier energy approach ω is estimated from the cross-spectrum of two image frames, in the Fourier differential method ω is a measurement from the three-dimensional Fourier transform of a sequence of images.

While Fleet and Jepson, and others [8,54,88,89,118], argue that the phase response is much more stable to noise than the brightness measures, they also point out that there are a number of complications in computing suitable phase information for three-dimensional signals. This can be largely overcome by decomposing the image into a set of two-dimensional signals, one spatial and one temporal. Quadrature filter pairs tuned for different spatial orientations are then used to estimate the image flow component in that direction. Each of the pairs consists of an even and odd equal magnitude impulse response function, specified by $Q_+(x,t)$ and $Q_-(x,t)$ respectively. The phase is given by,

$$\phi(x,t) = \tan^{-1} \frac{Q_-(x,t)}{Q_+(x,t)}, \tag{3.28}$$

from which the normal velocity component in the direction of the oriented filters can be computed,

$$\nu = \frac{\frac{\partial}{\partial x}\phi(x,t)}{\frac{\partial}{\partial t}\phi(x,t)}. \tag{3.29}$$

The overall velocity is composed by combining the oriented normal velocities.

Since phase signals are restricted to the $[-\pi, \pi)$ interval they are additively ambiguous by 2π. As such phase signals are discontinuous. Due to their inherent discontinuities, care must be taken in computing the partial derivatives of phase signals. Both Fleet and Jepson[88] and also Barron et al.[54] point out that this problem can be avoided by directly computing

the spatiotemporal phase gradients from the output of the quadrature filter pairs,

$$\frac{\partial \phi}{\partial x} = \frac{Q_+^2(x,t)\frac{\partial Q_-(x,t)}{\partial x} - Q_-^2(x,t)\frac{\partial Q_+(x,t)}{\partial x}}{Q_+^2(x,t) + Q_-^2(x,t)} \tag{3.30a}$$

$$\frac{\partial \phi}{\partial t} = \frac{Q_+^2(x,t)\frac{\partial Q_-(x,t)}{\partial t} - Q_-^2(x,t)\frac{\partial Q_+(x,t)}{\partial t}}{Q_+^2(x,t) + Q_-^2(x,t)}. \tag{3.30b}$$

Furthermore, this formulation of the phase gradients also eliminates the need to compute the inverse tangent trigonometric function. The resulting image flow being,

$$\begin{bmatrix} u \\ v \end{bmatrix} = -\sum_{\nu=0}^{N} \frac{Q_{+\nu}^2 \frac{\partial Q_{-\nu}}{\partial t} - Q_{-\nu}^2 \frac{\partial Q_{+\nu}}{\partial t}}{Q_{+\nu}^2 \frac{\partial Q_{-\nu}}{\partial x} - Q_{-\nu}^2 \frac{\partial Q_{+\nu}}{\partial x}} \tag{3.31}$$

where $\pm\nu$ is one of the $0...N$ quadrature pair filters. This is referred to as the Fourier phase method in the literature.

As can be seen from the preceding discussion both the Fourier energy and Fourier phase methods have advantages from utilizing phase information instead of image brightness. In the following sections comparison of the correlation and differential methods in the space-time domain are made instead of the Fourier domain. This is based on the poor performance of the Fourier energy approach observed by [54–56] and the additional complexity required by the Fourier phase method in comparison to its space-time equivalent.

3.5 Comparison of Methods

A brief comparison of Horn and Schunck's gradient algorithm against three variations of Singh and Allen's correlation algorithm is made. The variants include: the original method with a response distribution given by Eq.(3.5) and Eq.(3.6) for Laplacian based images, denoted method IF1; an alternative response function given by Eq.(3.7) and Laplacian imagery, denoted method IF2; and a method using response function Eq.(3.7) and brightness imagery, denoted IF3. No iterative smoothing of the image flow field is implemented for methods IF1 — IF3. Horn and Schunck's method is implemented with Gaussian spatiotemporal smoothing of $\sigma = 1.5$ and run for 100 iterations, it is referred to as method IF4.

A number of image sequences were considered and results for the original translating tree and diverging sequences are given in Table 3.1 and Table 3.2. These two sequences were chosen as good representative examples of all the sequences considered. The results shown

Method	Aperture Size	Mean	Median	Std. Dev.
IF1	3	0.5905	0.1897	2.48
		11.92°	0.24°	13.9°
	5	0.5490	0.1467	8.89
		11.46°	0.15°	14.2°
	7	0.5259	0.1093	0.90
		10.99°	0.09°	14.4°
	9	0.5509	0.0823	2.38
		10.75°	0.12°	15.0°
IF2	3	1.0036	0.7150	0.91
		19.31°	0.74°	12.7°
	5	0.9174	0.6229	8.80
		18.10°	0.60°	12.5°
	7	0.8498	0.5498	0.86
		17.14°	0.57°	12.5°
	9	0.7913	0.4956	0.85
		16.28°	0.42°	12.3°
IF3	3	0.5093	0.3066	0.59
		11.19°	0.30°	8.7°
	5	0.4638	0.2405	1.64
		10.19°	0.21°	8.4°
	7	0.3995	0.1922	0.68
		9.29°	0.16°	8.0°
	9	0.3395	0.1599	0.47
		8.42°	0.26°	7.5°
IF4	3	1.3801	1.1461	1.25
		23.98°	0.87°	13.4°
	5	0.6237	0.2645	0.78
		13.22°	0.28°	11.5°
	7	1.3698	1.1419	1.29
		23.59°	1.11°	13.0°
	9	3.3282	2.5698	3.52
		41.48°	1.94°	21.6°

Table 3.1: *Error metrics for the original translating tree image sequence. A comparison of Horn and Schunck's gradient based method (IF4) with three variations of Singh and Allen's correlation method (IF1 — IF3) for the original translating tree image sequence. The size of the correlation window and derivative kernel is given by the aperture size.*

Method	Aperture Size	Mean	Median	Std. Dev.
IF1	3	0.4096	0.2387	0.49
		20.92°	0.22°	12.4°
	5	0.4104	0.2319	0.49
		20.83°	0.29°	12.6°
	7	0.4117	0.2270	0.50
		20.76°	0.21°	12.8°
	9	0.4138	0.2227	0.51
		20.76°	0.19°	12.9°
IF2	3	0.2389	0.0508	0.47
		12.91°	0.05°	12.1°
	5	0.2349	0.0381	0.46
		12.32°	0.03°	12.5°
	7	0.2378	0.0309	0.47
		12.07°	0.06°	13.0°
	9	0.2414	0.0247	0.48
		11.95°	0.02°	13.4°
IF3	3	0.1283	0.0248	0.30
		8.87°	0.05°	9.0°
	5	0.1210	0.0171	0.31
		8.19°	0.01°	9.2°
	7	0.1167	0.0131	0.31
		7.73°	0.01°	9.4°
	9	0.1140	0.0112	0.31
		7.40°	0.01°	9.5°
IF4	3	0.1788	0.0653	0.30
		12.24°	0.12°	9.00°
	5	0.0889	0.0081	0.20
		6.94°	0.01°	7.72°
	7	0.1490	0.0405	0.27
		10.73°	0.04°	8.9°
	9	0.3238	0.1182	0.64
		16.43°	0.1570°	12.4°

Table 3.2: *Error metrics for the diverging tree image sequence. A comparison of Horn and Schunck's gradient based method (IF4) with three variations of Singh and Allen's correlation method (IF1 — IF3) for the diverging tree image sequence. The size of the correlation window and derivative kernel is given by the aperture size.*

are the mean, median and standard deviation of the square velocity error and the mean, median and standard deviation of the angular error metric. The metrics are computed for a number of different aperture sizes, which is the length of the derivative kernel in the case of IF4 or the length of one side of the correlation window in the case of IF1 — IF3.

The following observations can be made. Firstly, there is often a large difference between mean and median values with a correspondingly high standard deviation, a result which confirms comments made earlier in Section 2.9 that large errors tend to dominate mean error estimates of image flow. This is particularly evident in the case of the mean angular error metric.

Secondly, as expected for methods IF1 — IF3, the velocity estimate improves as the size of the correlation window increases and more data is used in the matching process. In the case of IF4 the velocity estimates initially improve as the size of the differential kernel is increased, but decreases as the kernel becomes too large and the computed differences become less representative of the instantaneous derivatives for the data.

Thirdly, although no one method is significantly superior, the results suggest that correlation based methods implemented with the raw image brightness measurements (IF3) perform better than those which use the Laplacian of image brightness (IF1 & IF2). Better results are possible for the correlation methods if smoothing is allowed, a result that is also true for Horn and Schunck's method (IF4). Figure 3.7 shows how the median square velocity error decreases with increasing number of iterations for IF4 computed using the translating square image sequence. The rate of convergence is controlled by γ. Unfortunately, smoothing is not always desirable, a point that is discussed further in the next section.

3.6 Implementation of Horn and Schunck's Algorithm

When implementing Horn and Schunck's algorithm a number of practical issues need to be considered: the spatial and temporal derivative masks, the size and form of the average velocity mask, acceptable values of γ and termination criteria for the iterative process. Appropriate choices here are often dependent on the nature of the problem and environment in which the algorithm is expected to operate.

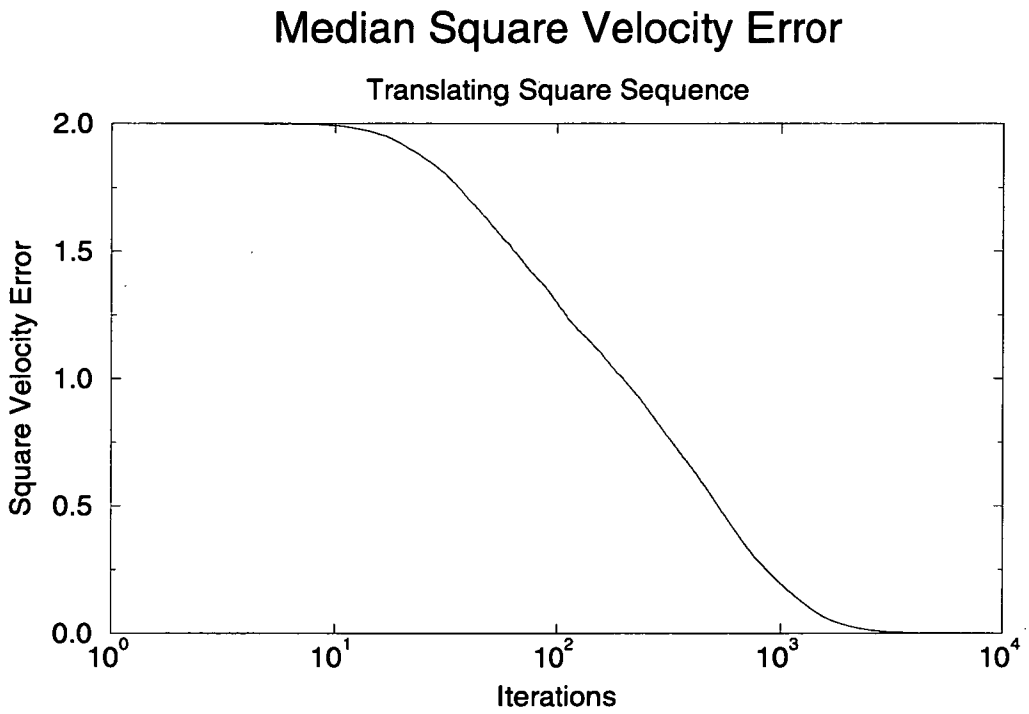


Figure 3.7: *Median square velocity of image flow computed using Horn and Schunck's method for the translating square sequence. Convergence controlled by* $\gamma = 0.5$.

In their original implementation Horn and Schunck[102] used forward differences based on a 2 point kernel:

-1	1

This approach calculates the velocity at points between pixels. However, by using central differences the gradient is computed at the pixel locations. Moreover, increasing the size of the derivative mask leads to significant improvement in the accuracy of the computed gradient, albeit at the expense of spatial resolution. The latter approach is the one adopted here, typically with 3, 5, 7 or 9 point masks before convolving with a smoothing kernel.

Following Barron et al.[54] implementation, the average velocity is computed using a 3×3 mask:

1/12	1/6	1/12
1/6	0	1/6
1/12	1/6	1/12

Care must be taken at the image boundaries, a point developed later.

Acceptable values for γ are implementation and scene dependent. Horn[7] argues that γ should be set according to the SNR of spatiotemporal gradients, with γ being small for accurate measurements and large if they are noisy. [1]

Few authors have discussed the issue of termination criteria of global techniques for estimating image flow. This is important as it is the iterative nature of these algorithms that give rise to their global behaviour. At any one instant in the iteration, the flow at a point is modified by only its immediate neighbours. With each successive iteration the influence of more distant points in the flow field will also cause change in the flow at the point, through the modification of its neighbours. Therefore, the smoothness of the flow field is controlled by both γ and the number of iterations, a result shown in [7, 8, 54, 102]. Although no attempt is made to resolve this issue, three consequences of it are demonstrated: a significant

[1] In Horn and Schunck[102], and Horn[7], the Lagrange multiplier was applied to the E^i term (cf. Eq.(3.13)) leading to the opposite requirements where γ is large for accurate measurements and small if they are noisy.

difference between the convergent global minimum and "sub-optimal" solutions, blurring of the flow field resolution, and the importance of boundary conditions on sensitivity.

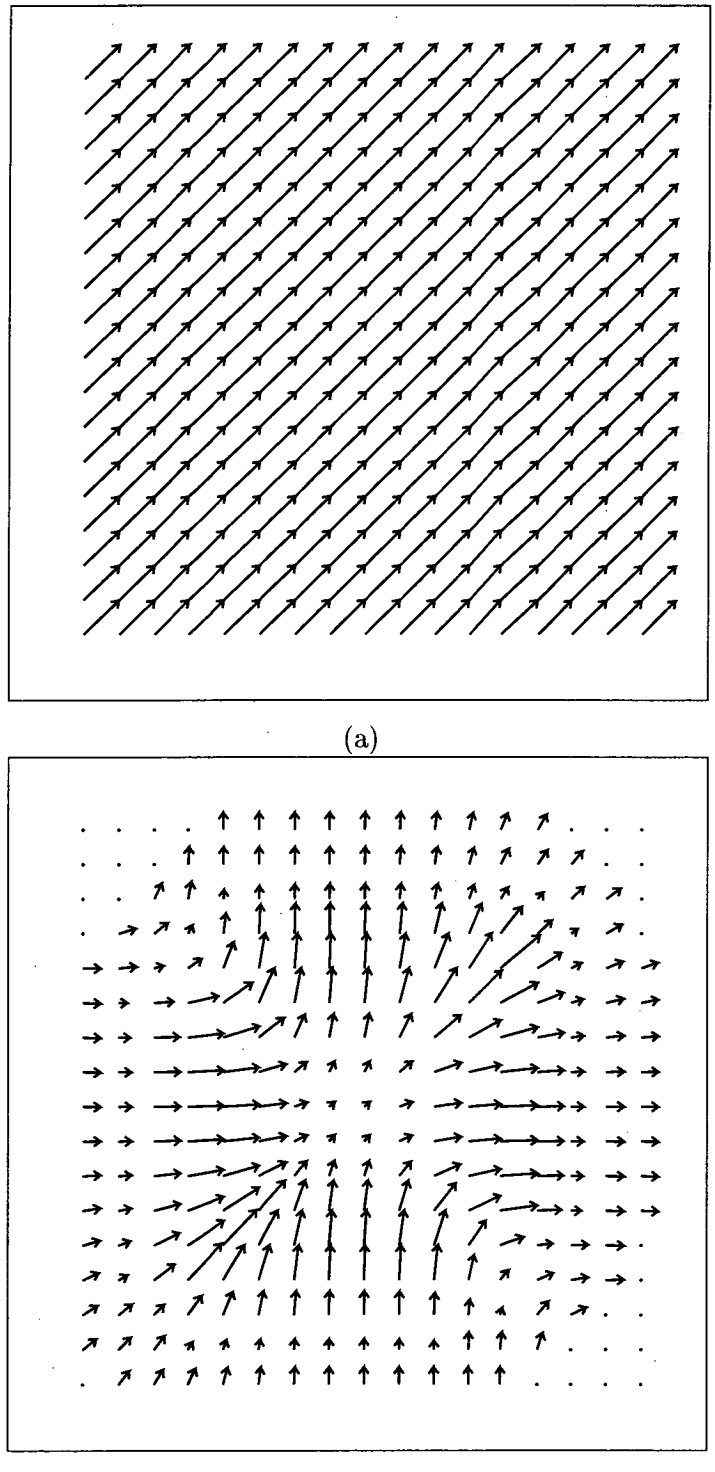
Consider the case of a dark translating square moving on a light background as shown in Figure A.1. The velocity field obtained from Horn and Schunck's algorithm when run to convergence (5000 iterations) is shown in Figure 3.8(a). In this case the velocity has been propagated across the entire field of view. This result differs from that presented by Barron et al.[56] where the iterations were terminated before the global minimum of Eq.(3.13a) was reached. Figure 3.8(b) shows the result when terminated early at the 100th iteration. While the constant image flow might at first sight appear surprising, it is in fact the true result of applying Horn and Schunck's algorithm to the data, because the algorithm has no way of discriminating between the foreground and background. The result is that the square and background are perceived to be part of the same object moving at a constant rate. Since this constant image flow is in fact perfectly smooth, it is inevitable that this corresponds to a lower E^s and consequently lower E^t for this solution than for that presented by Barron et al. and other similar non-uniform flows. The fact that most authors have obtained a flow described as,

> "varying from zero to the 'true' velocity at the squares corners,"

is simply an indication of the premature termination of the iterations. This ability to spread information (in particular erroneous information) without limit through bland image regions is a significant problem with Horn and Schunck's algorithm. It in part explains the sensitivity of Horn and Schunck's algorithm to errors in either derivatives or boundary conditions. Early termination of the iterations would reduce the sensitivity of the algorithm, but this has never been suggested as part of Horn and Schunck's algorithm.

With each successive iteration the flow field resolved by a global technique is diminished, a direct consequence of the blurring imposed by the smoothness constraint. This is evident in the case of a dark point moving on a light background as shown in Figure A.2. The velocity field obtained from our implementation for 3, 100 and 50000 iterations is shown in Figure 3.9. We observe how the velocity flow field has propagated across the entire field of view with no indication of the relative reliability of the estimates.

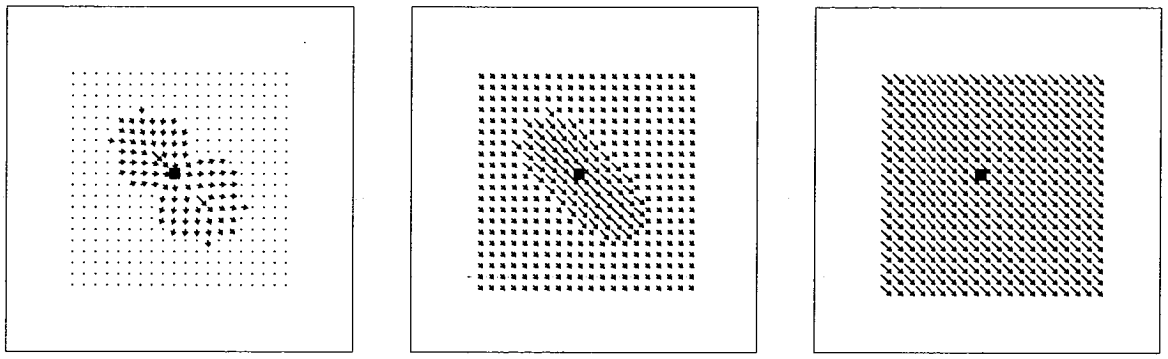
The extent of the loss of resolution in each iteration is also determined by the structure of



(a)

(b)

Figure 3.8: *Running Horn and Schunck's image flow technique (IF4) for: (a) 5000 iterations, (b) 100 iterations, on the moving square sequence shown in Figure A.1.*



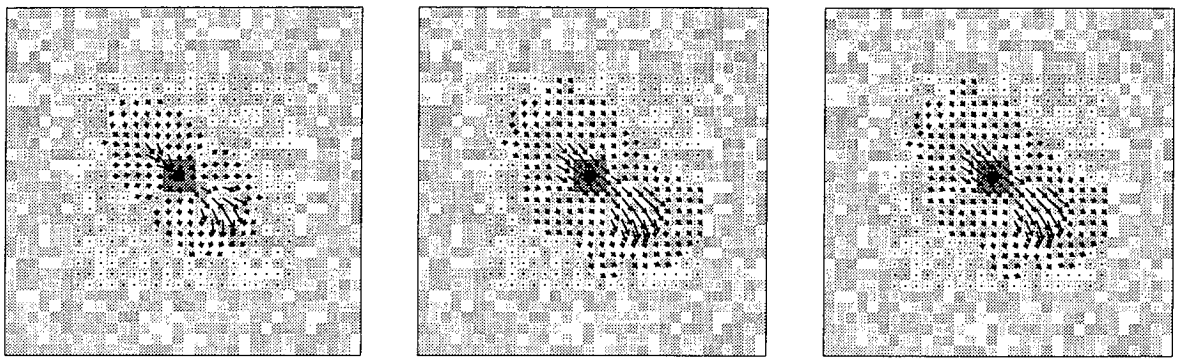
(a) (b) (c)

Figure 3.9: *Running Horn and Schunck's image flow technique (IF4) for: (a) 3 iterations, (b) 100 iterations, (c) and to convergence (50000 iterations) on the moving dot sequence shown in Figure A.2.*

the image sequence. In the dot sequence (Figure A.2) resolution is lost quickly, as illustrated in Figure 3.9 . When a stationary patterned background is present, as shown in the random dot sequence (Figure A.3), this is not the case, as can be seen in Figure 3.10. Identical parameters and kernel masks were used for both translating dot image sequences. We observe that the spread of the velocity field is substantially controlled by the information obtained from the static patterned background.

The feature of propagating the velocity field utilized by global smoothness algorithms becomes particularly relevant when applying smoothing at the image boundary. By convention, for a κ length derivative mask a border of $\kappa/2$ is required. In this border region the gradient cannot be computed accurately. The flow field averaging mask also requires a border region that is one half of its dimension. However, due to the algorithms iterative nature, the border size would increase at a rate of one half the averaging mask dimension per iteration. Such a characteristic is undesirable, so different boundary conditions are employed. Special care is now required, as in this case quite subtle changes in assumptions can result in dramatic changes in apparent performance, since the effects propagate throughout the entire image.

Tabulated in Table 3.3 are the results Barron et al.[56] obtained for the two synthetic translating and diverging tree image sequences. Their method effectively assumes the image flow is zero at the image boundary. Figure 3.11 shows the boundary velocity averaging masks they used. The sum of the coefficients for each of the masks do not add up to one. This assumes that points in the border region contribute, albeit with a value of zero, to



(a) (b) (c)

Figure 3.10: *Running Horn and Schunck's image flow technique (IF4) for: (a) 3 iterations, (b) 100 iterations, (c) and to convergence (50000 iterations) on the moving dot sequence shown in Figure A.3. Notice how the stationary patterned background substantially controls the level of velocity field spread.*

the average velocity. An alternative is to assume that image flow is continuous across the image boundary instead of zero, Figure 3.12 show the required set of boundary masks. In this case the velocity at points within the border are assumed to be equal to the average velocity computed at the boundary, leading to velocity masks which sum to unity. Given that the algorithms differ only in the assumptions of the image boundary conditions, the overall sensitivity of the image flow to local changes by Horn and Schunck method is readily apparent. This sensitivity was also noted by Barron et al.[56] with respect to errors in the computation of the intensity gradient functions. Flow fields for these two image sequences, using the smooth boundary conditions, are shown in Figure 1.6 and Figure 1.7.

3.7 Anisotropic Smoothing

Another interpretation can be put on the problem of the loss of resolution associated with smoothing of the velocity flow field. Real images, and real image sequences, contain discontinuities. These discontinuities can be due to: non-smooth variations in depth of the objects being viewed, objects moving at different speeds and directions, and so on. In fact all edge information seen in an image is due to some discontinuous property of the scene, and it is precisely at these points that we make our best estimates of the image flow. However, a smoothness constraint will try to eliminate such discontinuities and minimize change in the flow field. In the translating square and dot examples this is evident in the propagation of the flow field away from the discontinuous point.

Sequence and Method	Mean Angular Error	Median Angular Error	Standard Deviation
Translating Tree			
Barron Implementation (γ=0.5, 100 Iterations)	2.02°	—	2.27°
Modified Implementation (γ=0.5, 100 Iterations)	1.96°	0.004°	2.11°
Modified Implementation (γ=50, 5000 Iterations)	0.18°	0.0001°	0.15°
Diverging Tree			
Barron Implementation (γ=0.5, 100 Iterations)	2.55°	—	3.67°
Modified Implementation (γ=0.5, 100 Iterations)	2.22°	0.002°	2.21°
Modified Implementation (γ=50, 5000 Iterations)	1.50°	0.0004°	1.43°

Table 3.3: *Comparison of average angular error between variants of Horn and Schunck's algorithm, based on Fleet's error measure [4, 56].*

0	1/6
1/6	1/12

1/6	0	1/6
1/12	1/6	1/12

1/6	0
1/12	1/6

1/6	1/12
0	1/6
1/6	1/12

1/12	1/6
1/6	0
1/12	1/6

1/6	1/12
0	1/6

1/12	1/6	1/12
1/6	0	1/6

1/12	1/6
1/6	0

Figure 3.11: *Barron et al. set of average velocity boundary condition masks assumes non-smooth variation at the image boundary.*

0	2/5
2/5	1/5

1/4	0	1/4
1/8	1/4	1/8

2/5	0
1/5	2/5

1/4	1/8
0	1/4
1/4	1/8

1/8	1/4
1/4	0
1/8	1/4

2/5	1/5
0	2/5

1/8	1/4	1/8
1/4	0	1/8

1/5	2/5
2/5	0

Figure 3.12: *Alternate set of average velocity boundary condition masks assuming smooth variation.*

Horn[7], recognized this problem and proposed segmenting the flow field so that smoothing does not occur across such boundaries. However, he was aware this leads to a "chicken-and-egg problem":

> "If we have a good estimate of the image flow, we can look for places where it changes rapidly in order to segment the picture. On the other hand, if we could segment the picture well, we would produce a better estimate of the image flow."

Nagel [155–158], on the other hand argued that the dilemma is resolvable because the image can be segmented by examining the spatial gradients. Furthermore, he argued that a key requirement in preserving the estimates made in regions of discontinuity, is to impose an oriented smoothness constraint, as opposed to the isotropic constraint of Horn and Schunck. The aperture problem, refer to Eq.(3.11), clearly states that reliable estimates can only be made in direction of the spatial gradient (I_x, I_y). Little can be inferred in the orthogonal direction, unless the curvature of the intensity function is high at that point. Clearly, second order terms of the intensity function are required if a unique solution is to be found.

The oriented smoothness constraint requires minimizing change in the orthogonal direction of the intensity gradient,

$$\perp(\nabla I) = \begin{bmatrix} I_y \\ -I_x \end{bmatrix}, \tag{3.32}$$

because this is the direction of greatest uncertainty. In the approach of Singh and Allen[186] the direction of greatest uncertainty is equivalent to the maximum eigen direction (direction of least confidence from the spectral decomposition of the covariance matrix). The change of (u, v) in this direction is given by,

$$\Delta_\perp u = u_x I_y - u_y I_x, \tag{3.33a}$$

$$\Delta_\perp v = v_x I_y - v_y I_x, \tag{3.33b}$$

and hence the directional smoothness constraint is,

$$\text{Min}\{\Delta_\perp u^2 + \Delta_\perp v^2\} = \text{Min}\{(u_x I_y - u_y I_x)^2 + (v_x I_y - v_y I_x)^2\}. \tag{3.34}$$

Nevertheless, isotropic smoothing is still required to guarantee a solution in bland regions and it is weighted relative to the directional smoothness requirement by a constant γ^N

which is usually small. Hence Nagel's alternative smoothness constraint is,

$$E^{s_N} = \frac{1}{I_x^2 + I_y^2 + 2\gamma^N} \left((u_x I_y - u_y I_x)^2 + (v_x I_y - v_y I_x)^2 + \gamma^N (u_x^2 + u_y^2 + v_x^2 + v_y^2) \right). \tag{3.35}$$

The basis of Nagel's approach is two-fold, firstly the weighting of the smoothness requirement is modulated by $1/(I_x^2+I_y^2+2\gamma^N)$, and secondly a directional smoothness requirement $(u_x I_y - u_y I_x)^2 + (v_x I_y - v_y I_x)^2$ is introduced. The modulating term ensures that greater weighting is given to the smoothness constraint in regions of the image that are bland and therefore require velocity information from more distant neighbours to form an estimate. When high detail edge information is encountered the modulation term ensures that little weight is given to the smoothness constraint, thus preserving estimates made at these locations. This clearly meets the requirements specified by Horn[7] concerning the weighting functional γ with respect to the gradient SNR.

The error functional to be minimized is,

$$E^t = \iint \left(E^i + \zeta^2 E^{s_N} \right) dx\, dy. \tag{3.36}$$

Nagel gives the Euler-Lagrange equations for this minimization problem in (u, v) as,

$$0 = (I_x u + I_y v + I_t) I_x + \alpha^2 \frac{\nabla I^T \left(\begin{bmatrix} I_y y & -I_x y \\ -I_x y & I_x x \end{bmatrix} + 2 \begin{bmatrix} I_x x & I_x y \\ I_x y & I_y y \end{bmatrix} W \right) \begin{bmatrix} u_x \\ u_y \end{bmatrix}}{I_x^2 + I_y^2 + 2\gamma^N} - \alpha^2 \text{trace}(W \nabla^2 u) \tag{3.37a}$$

$$0 = (I_x u + I_y v + I_t) I_y + \alpha^2 \frac{\nabla I^T \left(\begin{bmatrix} I_y y & -I_x y \\ -I_x y & I_x x \end{bmatrix} + 2 \begin{bmatrix} I_x x & I_x y \\ I_x y & I_y y \end{bmatrix} W \right) \begin{bmatrix} v_x \\ v_y \end{bmatrix}}{I_x^2 + I_y^2 + 2\gamma^N} - \alpha^2 \text{trace}(W \nabla^2 v) \tag{3.37b}$$

where,

$$W = \frac{1}{I_x^2 + I_y^2 + 2\gamma^N} \begin{bmatrix} I_y^2 + \gamma^N & -I_x I_y \\ -I_x I_y & I_x^2 + \gamma^N \end{bmatrix}. \tag{3.37c}$$

Barron et al.[54] solve for (u, v) using the Gauss-Seidel iterative method,

$$\begin{bmatrix} u^{\tau+1} \\ v^{\tau+1} \end{bmatrix} = \begin{bmatrix} \xi(u^\tau) \\ \xi(v^\tau) \end{bmatrix} - \left(\frac{I_x \xi(u^\tau) + I_y \xi(v^\tau) + I_t}{I_x^2 + I_y^2 + \gamma^2} \right) \begin{bmatrix} I_x \\ I_y \end{bmatrix}, \tag{3.38}$$

where $\xi(u)$ and $\xi(v)$ are functions in first and second order spatial derivative terms of the image intensity and image flow.

In the experimental work of Willick and Yang[226] they compared the performance of Horn and Schunck[102] original algorithm to that of Nagel[158] and of Schunck[181]. They showed that the performance of Horn and Schunck's and also Nagel's methods were similar for the three synthetic test sequences considered, and performed much better than the others for the "small translational speed" test (speeds less than 1 pixel per frame). This is significant since aliasing may occur for speeds greater than 1 pixel per frame. Willick and Yang concluded:

> "Our results also reveal that under all situations Horn and Schunck's original motion constraint equation outperforms both Schunck's and Nagel's proposed alternatives. Although Nagel's is very close to Horn and Schunck's in two out of the three experiments, the added complexity of the equation makes it a much less attractive choice."

It is noted, without demonstration, that Nagel's formulation Eq.(3.38) will experience many of the same difficulties as Horn and Schunck's algorithm with respect to termination, blurring and the need for well formed boundary conditions. As shown earlier this may lead to quite undesirable results, as well as sensitivity to noise. Adding complexity to an algorithm is, in any case, a less appealing option. They are far more difficult to implement and tend to be less robust. Moreover, additional assumptions about the operating environment are usually required, thereby reducing the general application of the algorithm and introducing potential difficulties at the margins or extremes of its working conditions.

Other works of interest using global differential techniques include [47, 62, 78, 91, 126, 163, 165, 182, 190]. Nesi[163], investigated the problem of discontinuities and suggested a third constraint or penalty function based on Tikhonov regularization theory. Snyder[190], investigated the mathematical foundations of smoothness constraints, deriving both Horn and Schunck's, and Nagel's constraint equations.

We have observed the problems encountered by Horn and Schunck's algorithm which are largely attributable to its global iterative formulation. In the following chapter alternative non-iterative local differential techniques for computing the image flow field are investigated. These are low complexity algorithms which are based more closely on the underlying curvature information of the image sequence than the methods presented to date.

Chapter 4

Differential Image Flow

Horn and Schunck's global differential technique attempts to overcome the aperture problem through the use of a global smoothness constraint, which propagates information at a point to its neighbours. This can, however, lead to undesirable results. Even methods that attempt to preserve discontinuities, through the use of anisotropic smoothing functions, can produce results with a substantially reduced resolution. This is a fundamental characteristic of all methods which rely on assuming global properties of the image flow.

This chapter presents an investigation into local non-iterative differential techniques. The underlying goal of these techniques is to estimate a high accuracy flow field with the best possible resolution and density. This has the significant advantage that the impact of any measurement errors are localized.

There has often been a distinction drawn between first order methods based on solving Eq.(3.11) and second order methods which use Eq.(2.20), either independently, or in conjunction with Eq.(3.11). This distinction is somewhat artificial in that all first order methods eventually apply Eq.(3.11) over a finite region to obtain a solution. Unless there is some variation in the gradient over this region a solution for both components of the velocity is unobtainable. Thus the second order information is still implicitly required, and in this chapter it is shown that in practice the difference between a first order and second order algorithms is in the weightings applied to the conservation of intensity and gradient equations.

4.1 First Order Weighted Least Squares Technique

Instead of using a global smoothness constraint the error in Eq.(3.11) can be minimized over a localized aperture rather than the entire image, a method attributed by Barron et al.[56] to Lucas and Kanade [143]. The underlying assumption is that the velocity is constant over a finite region, unlike the assumption of a smooth variation over the entire image. The approach has the advantage that the effect of any local errors in either the data or model only affect the computed image flow over a finite region. The first order weighted least squares technique does this by minimizing the error metric,

$$\min_{u,v} \iint W(x-x_0, y-y_0)\big(I_x u + I_y v + I_t\big)^2 dx\, dy, \tag{4.1}$$

where $W(x,y)$ denotes a symmetric window function of finite size and usually designed to emphasize the data at the centre of the region relative to the edges. Similar approaches are suggested in [8, 79, 182, 185] and it should be noted that the exact shape of $W(x,y)$ is not critical.

When rearranged to form a set of equations, Eq.(4.1) becomes,

$$A^T W A \begin{bmatrix} u \\ v \end{bmatrix} = A^T W b \tag{4.2}$$

where the spatial and temporal gradient measurements are arranged,

$$A = \begin{bmatrix} I_x(x_1,y_1) & I_y(x_1,y_1) \\ I_x(x_2,y_2) & I_y(x_2,y_2) \\ \vdots & \vdots \\ I_x(x_n,y_n) & I_y(x_n,y_n) \end{bmatrix}, \quad \text{and,} \quad b = \begin{bmatrix} -I_t(x_1,y_1) \\ -I_t(x_2,y_2) \\ \vdots \\ -I_t(x_n,y_n) \end{bmatrix}.$$

The above equations can only be solved in situations where the matrix term $A^T W A$ is not singular. This implies that to produce a solution there must be a significant change in the gradients over the region defined by $W(x,y)$. This approach has the advantage over global methods in that errors in either the data or model will produce only localized errors in the computed image flow. For the purpose of latter discussions this method is referred to as IF5.

4.2 First Order Functional Method

In their recent paper Sobey and Srinivasan[191], proposed an algorithm in which the image flow was determined by first convolving the image with two independent functions, denoted

by $g(x,y)$ and $h(x,y)$, to produce two new functions,

$$O^g(x,y) = I(x,y) \odot g(x,y), \tag{4.3a}$$

$$O^h(x,y) = I(x,y) \odot h(x,y). \tag{4.3b}$$

They then showed that the velocity can be obtained by solving the system of equations,

$$\begin{bmatrix} O^g_x & O^g_y \\ O^h_x & O^h_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} O^g_t \\ O^h_t \end{bmatrix}. \tag{4.4}$$

Perrone[172] proposed a similar approach using the sum of a number of directionally tuned filters.

The functions $g(x,y)$ and $h(x,y)$ can be chosen almost arbitrarily provided they are linearly independent. The results presented by Sobey and Srinivasan in [191] used,

$$g(x,y) = (1+\cos(\pi y))\sin(\pi x) \qquad -1 \leq \tfrac{2x}{(N_x-1)}, \tfrac{2y}{(N_y-1)} \leq 1, \tag{4.5a}$$

and,

$$h(x,y) = (1+\cos(\pi x))\sin(\pi y) \qquad -1 \leq \tfrac{2x}{(N_x-1)}, \tfrac{2y}{(N_y-1)} \leq 1. \tag{4.5b}$$

These are respectively the partial derivatives of the raised cosine function,

$$s(x,y) = (1+\cos(\pi x))\,(1+\cos(\pi y))\,, \tag{4.5c}$$

with respect to x and y and where $N_x \times N_y$ denotes the size of mask over which the functions are defined. A plot of the raised cosine function, $s(x,y)$, is shown in Figure 4.1. As can be seen the raised cosine function is similar to the Gaussian smoothing function, Figure 4.2. Hence, for this choice of $g(x,y)$ at least,

$$\begin{aligned} g(x,y) &= D_x \odot s(x,y) \\ \therefore \quad O^g_x &= D_x \odot D_x \odot s(x,y) \odot I(x,y) \\ &= s(x,y) \odot I_{xx}(x,y). \end{aligned} \tag{4.5d}$$

Similarly O^h_x, O^h_y, O^g_y are also smoothed second order derivatives of the image intensity function. The results are equivalent to those that would have been obtained by applying the second order equations given in Eq.(2.20), to data smoothed by the weighting function in Eq.(4.5c). Consequently it can be argued that the more robust performance of this method

when compared to other second order techniques is due to smoothing of the derivative calculation and not because this is inherently a first order method. As shown earlier in Section 2.8 regularization is a necessary requirement for computing derivatives in the presence of noise. Sobey et al. further developed their technique to estimate first order terms of the image flow [192].

4.3 Second Order Techniques

Rewriting Eq.(2.20a) and Eq.(2.20b) in matrix form yields the common second order method for the calculation of the image flow,

$$\begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_{xt} \\ I_{yt} \end{bmatrix}. \tag{4.6}$$

This approach is discussed by Nagel [157], Verri and Poggio [71,152,212–214] and implicitly used by Sobey and Srinivasan[191]. Although the approach of Sobey and Srinivasan in its most general form can differ from a pure second order method, it reduces in the examples presented in [191] to a second order method where the data has first been smoothed (which is a necessary requirement). This method is referred to as IF7.

An obvious extension to this common second order approach, and attributed by Nagel[157] to both Haralick and Lee, and, Tretiak and Pastor[207] (see also [3,6,8]), is to incorporate the first order equation to form an over-determined system of equations,

$$\begin{bmatrix} I_x & I_y \\ I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}^T L \begin{bmatrix} I_x & I_y \\ I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_x & I_y \\ I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}^T L \begin{bmatrix} I_t \\ I_{xt} \\ I_{yt} \end{bmatrix}. \tag{4.7}$$

L is a weighting function, and in the cases defined by both Haralick and Shapiro[6], and, Tretiak and Pastor[207] it is defined as the identity matrix I. However, when solving over-determined least squares problems of this form it is usual to attach different weights with the different equations,

$$L = \begin{bmatrix} \lambda_0 & 0 & 0 \\ 0 & \lambda_1 & 0 \\ 0 & 0 & \lambda_1 \end{bmatrix}, \tag{4.8}$$

where the matrix containing λ_0 and λ_1 provides the differential weighting on the equations.

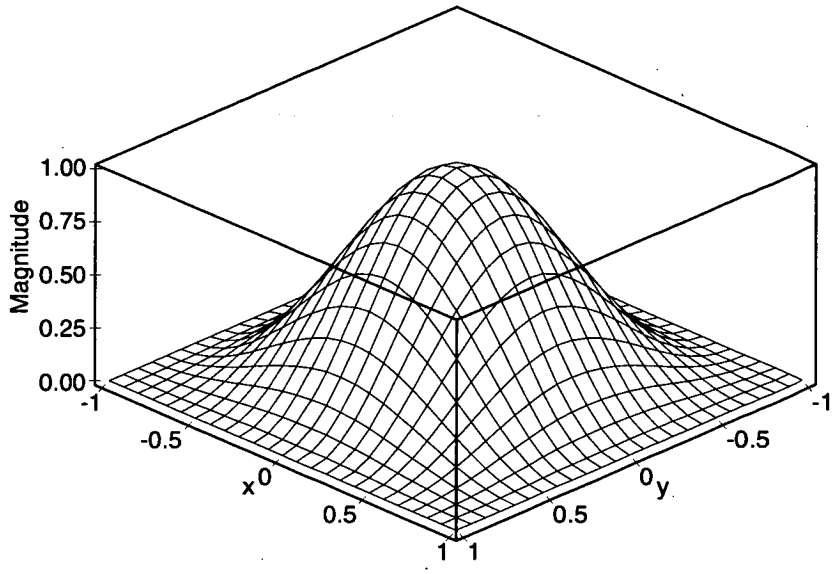


Figure 4.1: *Sobey and Srinivasan's function is equivalent to a two-dimensional smoothing function.*

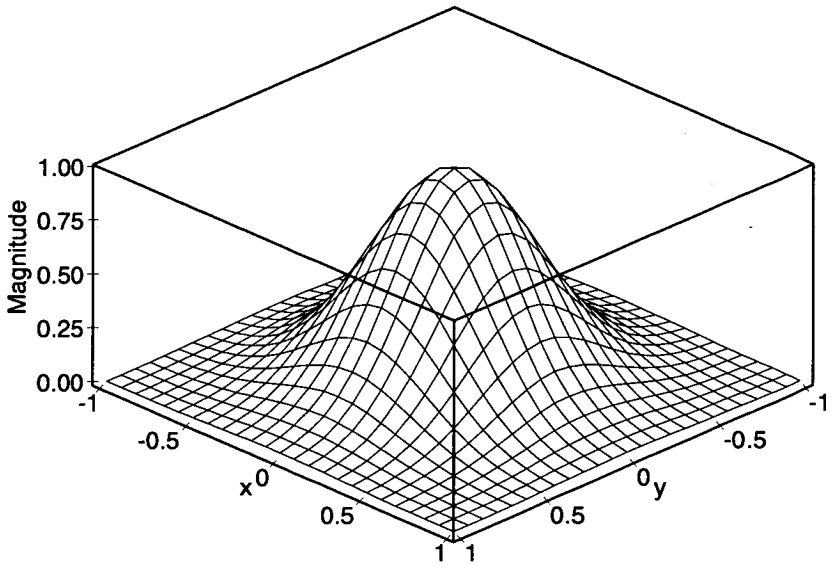


Figure 4.2: *A two-dimensional Gaussian smoothing function.*

By expanding and letting $\lambda = \lambda_0/\lambda_1$ the velocities are thus found by solving,

$$-\begin{bmatrix} \lambda I_x I_t + I_{xx} I_{xt} + I_{xy} I_{yt} \\ \lambda I_y I_t + I_{xy} I_{xt} + I_{yy} I_{yt} \end{bmatrix} = \begin{bmatrix} \lambda I_x^2 + I_{xx}^2 + I_{xy}^2 & \lambda I_x I_y + I_{xx} I_{xy} + I_{yy} I_{xy} \\ \lambda I_x I_y + I_{xx} I_{xy} + I_{yy} I_{xy} & \lambda I_y^2 + I_{yy}^2 + I_{xy}^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tag{4.9}$$

an approach referred to as an augmented second order solution, method IF6. In most cases an appropriate weight matrix would usually place greater emphasis on the conservation of intensity equation with respect to the conservation of gradient intensity equations, i.e. $\lambda > 1$.

4.4 Equivalence of First and Second Order Techniques

It is possible to demonstrate that the first order weighted least squares solution to the image flow constraint equation is equivalent to the second order solution for small apertures. Consider the matrix expansion of $A^T W A$,

$$\begin{bmatrix} \iint W(x - x_0, y - y_0) I_x^2 dx\, dy & \iint W(x - x_0, y - y_0) I_x I_y dx\, dy \\ \iint W(x - x_0, y - y_0) I_x I_y dx\, dy & \iint W(x - x_0, y - y_0) I_y^2 dx\, dy \end{bmatrix}, \tag{4.10}$$

and the first element of the matrix, $\iint W(x - x_0, y - y_0) I_x^2(x, y) dx\, dy$. Assuming the aperture is restricted to a size where the surface $I(x, y)$ can only be described adequately by a second order Taylor series, requires discarding terms higher than first order. For the expansion of $W(x - x_0, y - y_0) I_x^2(x, y)$ about (x_0, y_0) this yields,

$$\iint W\ (x - x_0, y - y_0)\Big(I_x(x_0, y_0) + I_{xx}(x_0, y_0)(x - x_0) + I_{xy}(x_0, y_0)(y - y_0)\Big)^2 dx\, dy, \tag{4.11a}$$

which can be expanded to,

$$\begin{aligned} \iint W\ (x - x_0, y - y_0)\Big(& I_x^2(x_0, y_0) + (I_{xx}(x_0, y_0)(x - x_0))^2 \\ & + (I_{xy}(x_0, y_0)(y - y_0))^2 + 2 I_x(x_0, y_0) I_{xx}(x_0, y_0)(x - x_0) \\ & + 2 I_x(x_0, y_0) I_{xy}(x_0, y_0)(y - y_0) \\ & + 2 I_{xx}(x_0, y_0) I_{xy}(x_0, y_0)(x - x_0)(y - y_0)\Big) dx\, dy. \end{aligned} \tag{4.11b}$$

If $W(x,y)$ is chosen to be a symmetric function all terms involving $(x-x_0)$, $(y-y_0)$ or $(x-x_0)(y-y_0)$ integrate to zero. Using the notation,

$$\lambda_0 = \iint W(x,y)dx\,dy, \tag{4.12a}$$

and,

$$\begin{aligned}\lambda_1 &= \iint W(x,y)x^2dx\,dy \\ &= \iint W(x,y)y^2dx\,dy,\end{aligned} \tag{4.12b}$$

we can thus state,

$$\iint W(x-x_0,y-y_0)I_x^2(x,y)dx\,dy \approx \lambda_0 I_x^2 + \lambda_1 I_{xx}^2 + \lambda_1 I_{xy}^2, \tag{4.13}$$

where it is understood that the derivatives are computed at the point (x_0,y_0). Hence Eq.(4.2) can be rewritten as,

$$-\begin{bmatrix} \lambda I_x I_t + I_{xx}I_{xt} + I_{xy}I_{yt} \\ \lambda I_y I_t + I_{xy}I_{xt} + I_{yy}I_{yt} \end{bmatrix} = \begin{bmatrix} \lambda I_x^2 + I_{xx}^2 + I_{xy}^2 & \lambda I_x I_y + I_{xx}I_{xy} + I_{yy}I_{xy} \\ \lambda I_x I_y + I_{xx}I_{xy} + I_{yy}I_{xy} & \lambda I_y^2 + I_{yy}^2 + I_{xy}^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tag{4.14}$$

where $\lambda = \lambda_0/\lambda_1$. The system of equations that is solved results from a weighted sum of terms involving both first and second order derivatives of the image intensity. Hence when limited to our ability to resolve higher order derivative terms of the image intensity, the first order weighted least squares technique effectively reduces to an augmented second-order technique. When this limit is not reached it includes the effect of higher order derivatives and can thus be expected to be more robust in performance when compared with the other derivative methods.

4.5 Systematic Bias Errors

The least squares estimate of Eq.(4.2) can be rewritten to take into account measurement error based on the finite difference approximation,

$$A = \bar{A} + A^e, \tag{4.15a}$$

$$b = \bar{b} + b^e, \tag{4.15b}$$

$$[u\,v]^T = [\bar{u}\,\bar{v}]^T + [u^e\,v^e]^T, \tag{4.15c}$$

where b^e is the error in estimating the exact temporal gradients $\bar{b}$ and A^e is a matrix of error terms associated with estimating the exact spatial gradients $\bar{A}$. Recall that A and b are the actual gradients computed from the image sequence intensity measurements. Then by algebraic manipulation an expression for the velocity error can be derived,

$$
\begin{aligned}
& A^T W A[u\ v]^T = A^T W b \\
\Leftrightarrow \quad & A^T W A([\bar{u}\ \bar{v}]^T + [u_e\ v_e]^T) = A^T W b \\
\Leftrightarrow \quad & A^T W A[u_e\ v_e]^T = A^T W \left(b - A[\bar{u}\ \bar{v}]^T\right) \\
& \qquad = A^T W \left(\bar{b} + b_e - \bar{A}[\bar{u}\ \bar{v}]^T - A_e^T[\bar{u}\ \bar{v}]^T\right) \\
\Longrightarrow \quad & \begin{bmatrix} u_e \\ v_e \end{bmatrix} = (A^T W A)^{-1} A^T W \left(b_e - A_e \begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix}\right),
\end{aligned}
\tag{4.16}
$$

where $\bar{b} - \bar{A}[\bar{u}\ \bar{v}]^T = 0$.

The usual assumption made in least squares analysis is that the 'measurement' matrix A is made without error and the errors in the 'observation' vector b are Gaussian distributed with zero mean. For this problem the assumption of an error free A is unrealistic, it is corrupted with additive sensor noise in the image capture phase and further error occurs in the computation of the gradients, for example by finite difference approximation. However, assuming the errors in computing the gradients are far more significant than all other sources, the matrix A^e can then be approximated by,

$$
A^e = \Upsilon(N)\delta_s^{N-1} X, \tag{4.17a}
$$

and,

$$
X = \begin{bmatrix}
\frac{\partial^N I(x_1,y_1)}{\partial x^N} & \frac{\partial^N I(x_1,y_1)}{\partial y^N} \\
\frac{\partial^N I(x_2,y_2)}{\partial x^N} & \frac{\partial^N I(x_2,y_2)}{\partial y^N} \\
\vdots & \vdots \\
\frac{\partial^N I(x_n,y_n)}{\partial x^N} & \frac{\partial^N I(x_n,y_n)}{\partial y^N}
\end{bmatrix}, \tag{4.17b}
$$

where δ_s is the spatial sampling size and the gradients are approximated by an N-th order central difference, see Section 2.7. Similarly, b^e will be a sum of the errors associated with computing the temporal gradients, plus a random noise component r which is uncorrelated with the temporal gradients or A. The error term r, as is standard practice for least squares analysis, is usually added to the observation vector b [16,18–20,23–27,30,93]. Denoting the

temporal sampling size as δ_t then,

$$b_e = \Upsilon(N)\delta_t^{N-1} \begin{bmatrix} \frac{\partial^N I(x_1,y_1)}{\partial t^N} \\ \frac{\partial^N I(x_2,y_2)}{\partial t^N} \\ \vdots \\ \frac{\partial^N I(x_n,y_n)}{\partial t^N} \end{bmatrix} + r. \tag{4.17c}$$

The distance between samples in the temporal and spatial domains directions are δ_t and δ_s respectively. Recall from Eq.(3.23) that the temporal and spatial sampling are related by the velocity of of the image point, hence,

$$\begin{aligned} \delta_t &= \frac{1}{\delta_s}\bar{u} + \frac{1}{\delta_s}\bar{v} \\ &= |\bar{u}\ \bar{v}|, \end{aligned} \tag{4.18}$$

under the condition that $\delta_s = 1$ pixel/sample and the velocity is given in pixel/frame. Similarly the higher order temporal derivatives are related to the higher order spatial derivatives by,

$$\frac{\partial^N I}{\partial t^N} = \frac{1}{|\bar{u}\ \bar{v}|} \begin{bmatrix} \frac{\partial^N I}{\partial x^N} \\ \frac{\partial^N I}{\partial y^N} \end{bmatrix}^T \begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix}, \tag{4.19}$$

which shows that the temporal derivative is equivalent to the line derivative through the two-dimensional space of the image plane in the direction given by the velocity [28]. This expression is valid provided that differential spatial terms higher than N are insignificant, an assumption made in Eq.(2.36c). The spatial and temporal derivative errors can now be rewritten as,

$$A_e = \Upsilon(N)X, \tag{4.20a}$$

$$b_e = \Upsilon(N)|\bar{u}\ \bar{v}|^{N-2}\left(X\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} + r\right), \tag{4.20b}$$

and by substitution into Eq.(4.16) the velocity error becomes,

$$\begin{bmatrix} u_e \\ v_e \end{bmatrix} = \Upsilon(N)\left(|\bar{u}\ \bar{v}|^{N-2} - 1\right)(A^TWA)^{-1}A^TW\left(X\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} + r\right) \tag{4.20c}$$

The following points are apparent from Eq.(4.20c). Firstly, the error is dependent on the true optical flow $(\bar{u}, \bar{v})$. Secondly, the error is dependent upon the size of the higher order

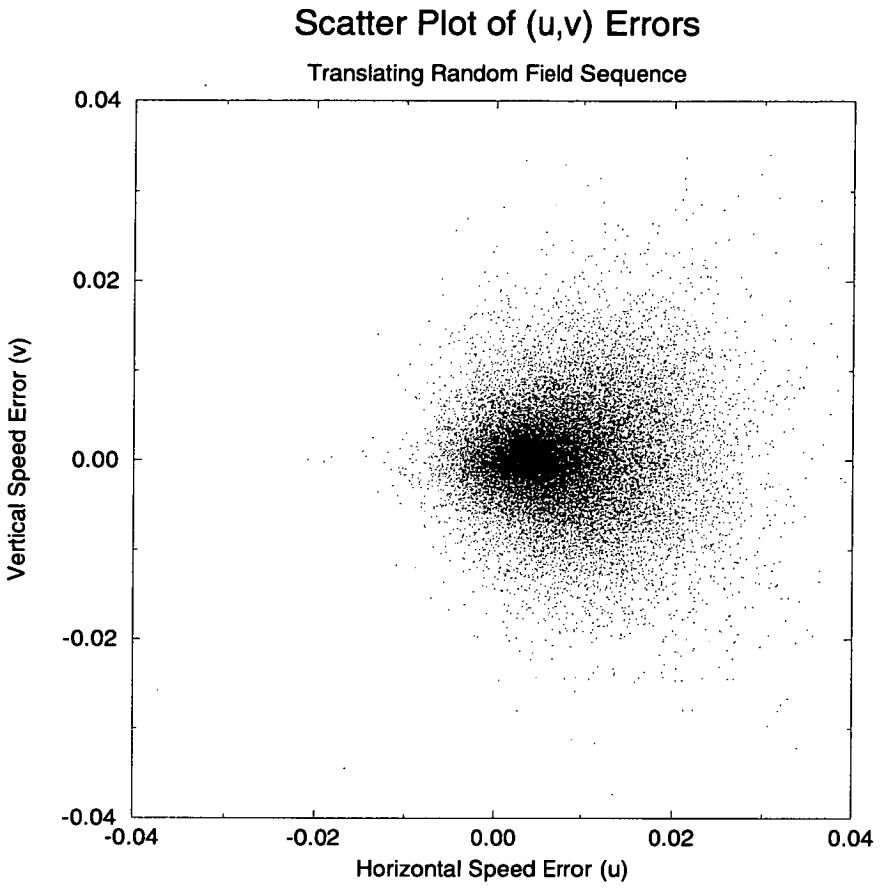


Figure 4.3: *Scatter plot of speed errors for the* $(0.3, 0)$ *translating random field sequence, see Figure A.4, computed using the IF5 algorithm.*

spatial gradients and the length of the derivative kernel, where the magnitude of X can reasonably be expected to diminish for larger difference kernels. Thirdly, the expectation of Eq.(4.16) is non-zero except where $|\bar{u}\ \bar{v}| = 0$ or 1, and hence the image flow estimate is systematically biased. This is readily apparent in Figures 4.3 — 4.5, which show the errors in image flow estimates for all three methods IF5, IF6 and IF7 computed using the uniformly distributed translating (0.3 pixel/frame) random field image sequence, Figure A.4. The bias in the direction of motion is clearly visible with the variance also being greater in this direction. These results can also be predicted from Eq.(4.20c), where the projection operation of $(A^T W A)^{-1} A^T W$ ensures that the bias is weighted in the direction of the optical flow with the a larger error variance.

The systematic bias is expected due to the correlation between X and A, which results from the correlation of temporal and spatial gradients. Hence, all methods of computing image flow based on gradients will posses a systematic bias in their estimates, a result

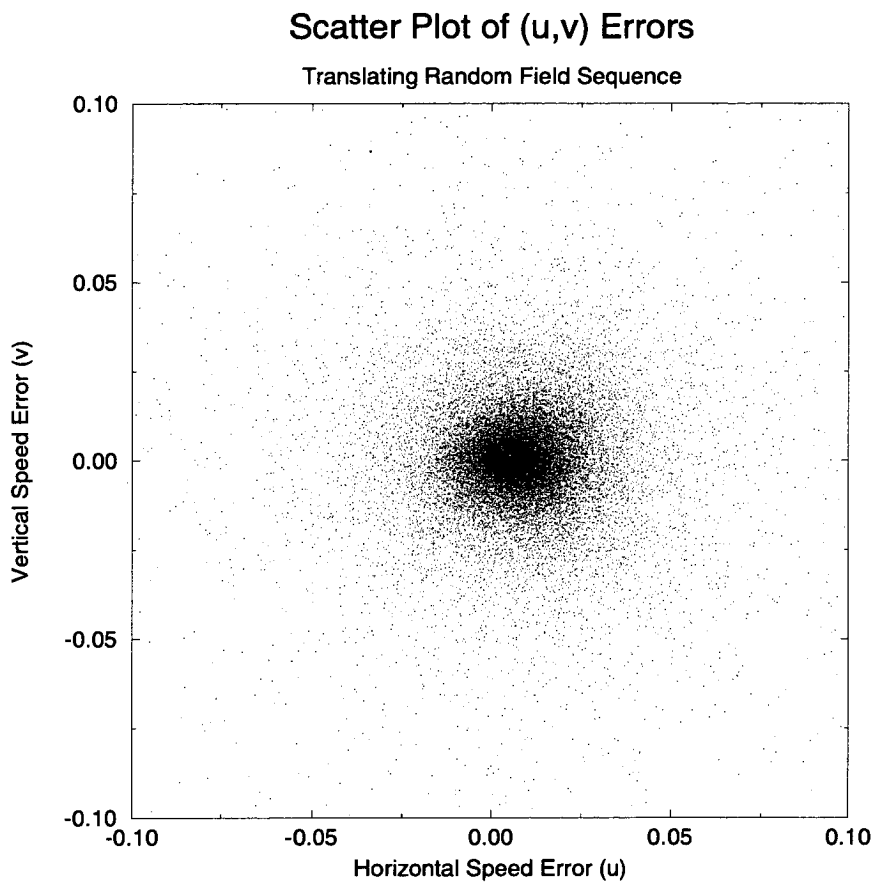


Figure 4.4: *Scatter plot of speed errors for the* $(0.3, 0)$ *translating random field sequence, see Figure A.4, computed using the IF6 algorithm.*

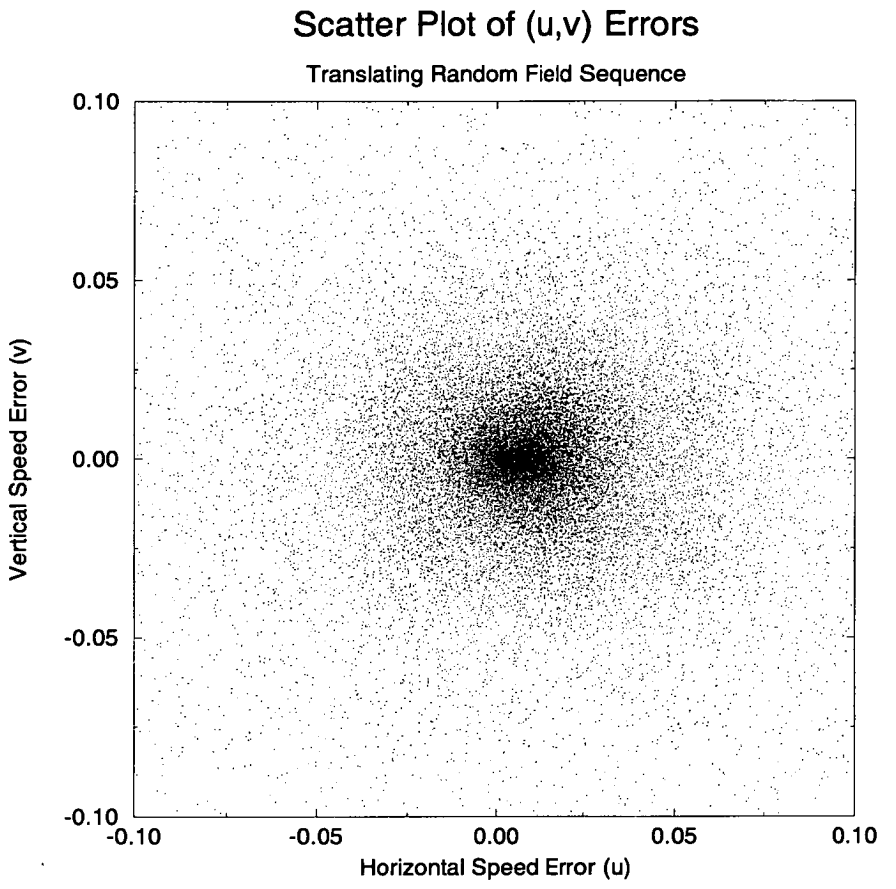


Figure 4.5: *Scatter plot of speed errors for the* $(0.3, 0)$ *translating random field sequence, see Figure A.4, computed using the IF7 algorithm.*

demonstrated by Figures 4.3 — 4.5. By way of a further demonstration, the velocity error of a noiseless translating greyscale corner was analytically analyzed using the mathematical package *Maple* [22]. The gradients were computed using a 5 point central difference kernel with varying degrees of Gaussian smoothing $\alpha = 1 \ldots 2.5$. The resulting squared velocity error shown in Figure 4.6 is computed for an optical flow of $u = 0 \ldots 1.2, v = 0$ at the pixel point $(0, 0)$ at time 0. This point was chosen as it corresponds to the point where the error in the gradient measures are greatest. It is apparent that the velocity error varies smoothly with speed, increasing monotonically for speeds greater than 1 pixel/frame and, as expected, having global minimum points corresponding to speeds of 0 and 1 pixel/frame. The velocity error decreases smoothly as the width of the smoothing kernel increases (the regularization increases).

It is possible, by using different spatial and temporal kernels, to change the speed at which a perfect estimate in a noiseless environment is made. This factor could prove useful in implementing a coarse to fine estimation or multi-resolution process, an issue discussed in Chapter 7. However, for simplicity in this work the same differential kernel was chosen for both spatial and temporal derivatives.

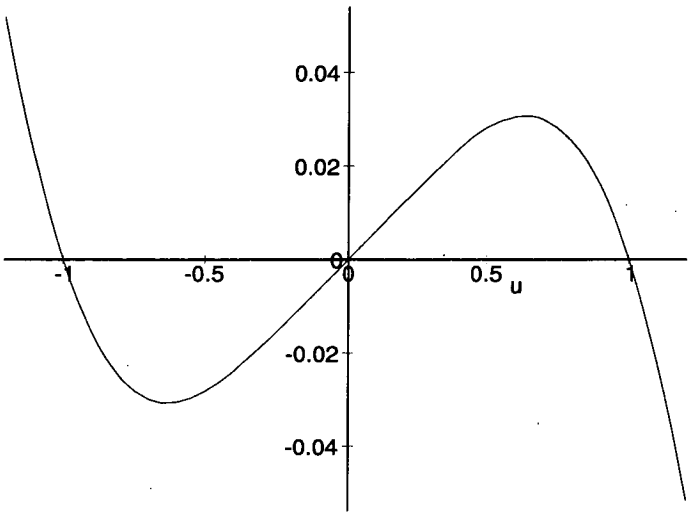
4.6 Noise Analysis

In order to quantify accurately the effects of smoothing and velocity on the estimated image flow, translating random field image sequences were synthesized in the Fourier domain. This allows accurate sub-pixel movements between frames to be produced. The n^{th} frame of a sequence is generated using,

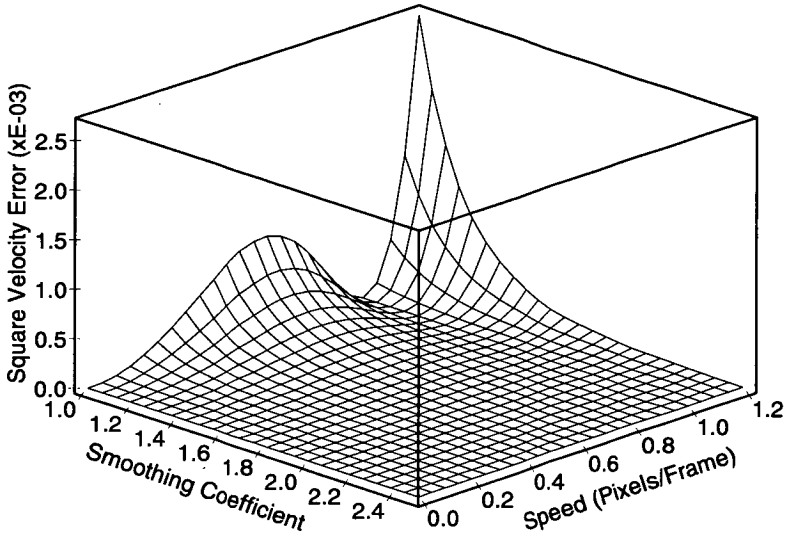
$$I^n(x, y) = I^0(x - un, y - vn) = \iint \mathcal{I}(k_x, k_y) e^{j(k_x un + k_y vn)} dk_x dk_y, \tag{4.21}$$

where $\mathcal{I}(k_x, k_y)$ is the Fourier transform of the initial frame $I^0(x, y)$, (k_x, k_y) the spatial frequency components, (u, v) is the velocity in pixels per frame and n is the frame number. The optical flow is thus identical for each point in the frame, irrespective of its relative depth. This is physically reasonable provided the field of view is very shallow, i.e. the relative distance between the closest and furtherest points from the sensor focal point is small compared to the average distance of points from the sensor. A total of 17 frames are synthesized in each sequence, with velocities ranging between 0 and 1.8 pixels/frame.

The first test conducted was to examine the effects of uncorrelated Gaussian noise on the



(a)



(b)

Figure 4.6: *Square velocity error associated with a noiseless translating greyscale corner for the point* $(0, 0)$ *shown in Figure 2.17(b). Systematic errors in estimating the velocity due to finite difference approximation of the gradient functions are illustrated as: (a) a function of velocity for a smoothing of* $\alpha = 1$*, (b) a function of smoothing and velocity.*

translating random field sequence when the motion is chosen to be equal to 1 pixel/frame across the entire image. As noted earlier, this motion ensures that the errors in estimating the velocity field are entirely due to the noise and not modeling (bias) errors. From Eq.(2.44) we see that the noise in the gradients is proportional to the noise of the image intensities. Hence, we would expect the velocity estimate errors to also be proportional to the image intensity noise. This is confirmed in Figure 4.7 where log of median square error has been plotted as a function of the signal to noise ratio of the image intensity. We observe that as the SNR increases from 0dB the log(median square error) decreases proportionally at a rate independent of the particular spatiotemporal smoothing. When the SNR drops below 0dB the estimated velocity is dominated by the additive uncorrelated noise and approaches the value that would be obtained if all frames were uncorrelated both spatially and temporally. Finally, for very high SNR, above 100dB, the results are limited by the numerical precision of the computation.

Figure 4.8 shows the effect of varying the spatiotemporal smoothing applied to the noise corrupted translating random field image sequence. We observe that as the smoothing increases, defined by α, the median square error initially decreases as the SNR of the derivative measures improve. The median square error then increases as the amount of smoothing becomes excessive. This behaviour is attributable to the increasing numerical errors associated with computing very small derivatives being amplified by poor conditioning of the matrix inverse.

4.7 Comparison of Methods

Performance comparisons of the first order weighted least squares method and the two second order methods are made on the basis of equal size apertures for the algorithms. For the second order and augmented second order techniques the aperture size is defined to be the size of the smoothed second order derivative mask. For the first order weighted least squares method the aperture size is defined by the size of the smoothed first order derivative mask added to the area $W(x, y)$ over which Eq.(4.1) is defined. In our implementation of the first order weighted least squares algorithm we use the Gaussian weighting function used in [56], with a spatial extent of $4\beta+1$ where β^2 is the variance of the weighting function. The mask length for the differential kernels and β values for a number of different presmoothed aperture sizes are given in Table 4.1. Four test image sequences are employed for the purpose of making quantitative comparisons, namely: translating random field sequences

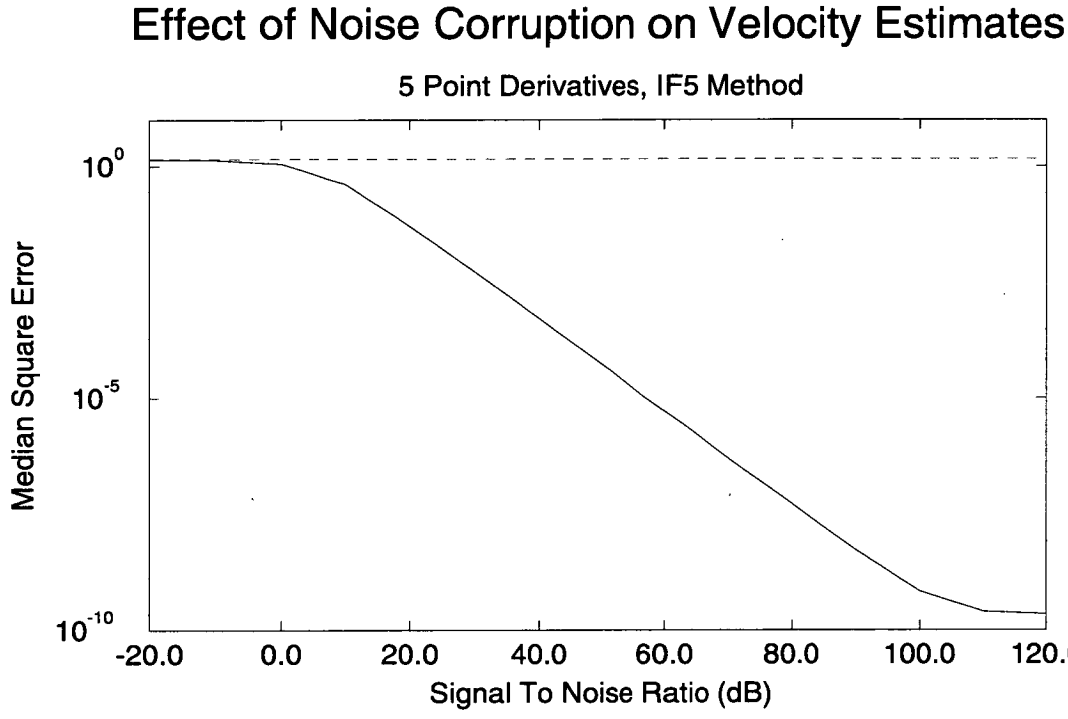


Figure 4.7: *Effect of additive uncorrelated noise on the computed velocity field errors. Solid line: noise added to the translating random field sequence, $\sigma_r = 1.0$. Dashed line: uncorrelated noise added to a blank field.*

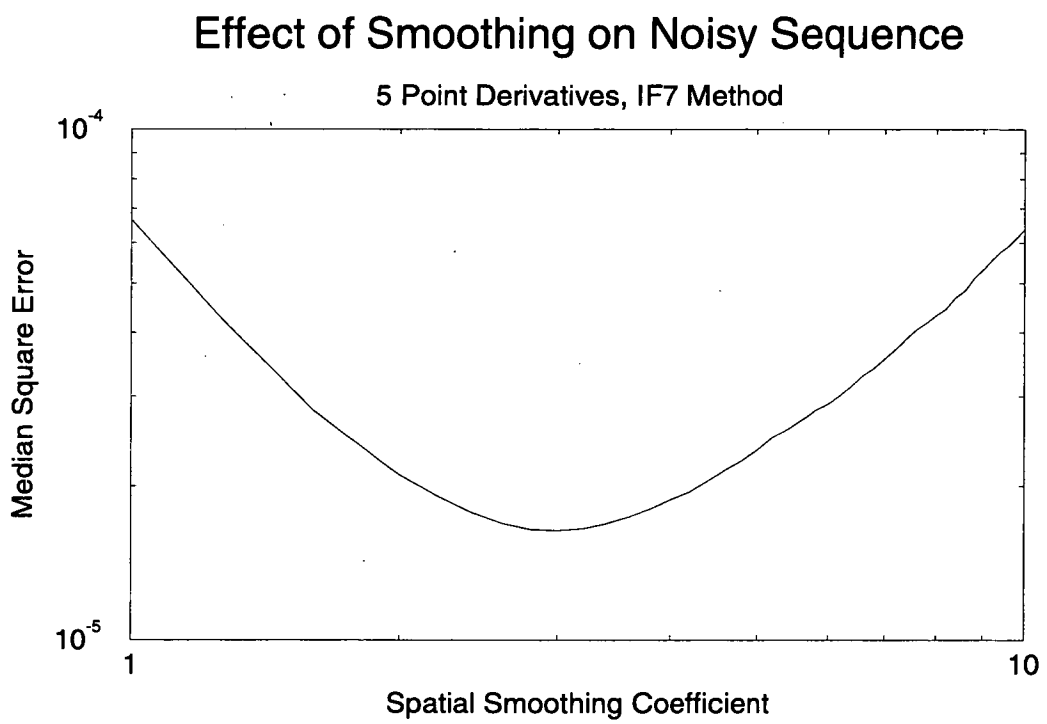


Figure 4.8: *Effect of smoothing on computed velocity field errors for a noisy translating random field, $\sigma_n = 10^{-3}$ (SNR of 30dB). $\beta = 2.0$. Expected velocity is 1 pixel/frame.*

Aperture Size	Derivative Mask Size		β
	First	Second	
5	3	5	0.5
9	5	9	1.0
13	7	13	1.5
17	9	17	2.0

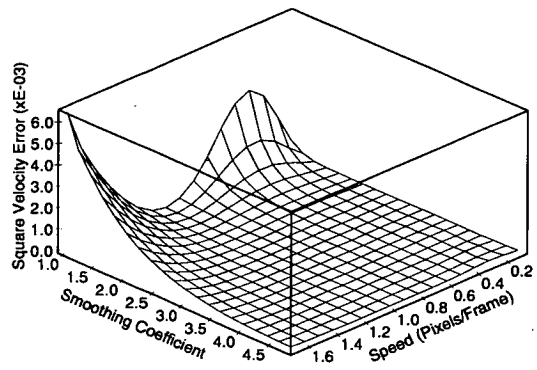
Table 4.1: *Relating aperture size to the size of the derivative kernel and the first order least squares weighting function (β).*

Figure A.4, modified translating tree sequences, the original translating tree Figure A.5, and the diverging tree sequences Figure A.6. The modified tree sequence was also generated using the Fourier phase shifting technique, Eq.(4.21), to allow accurate sub-pixel optical flow fields to be generated.

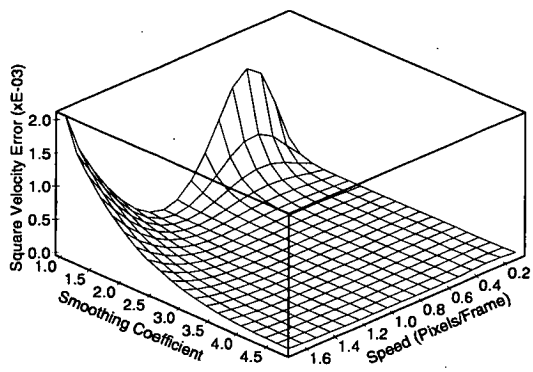
Figure 4.9 and Figure 4.10 illustrate the speed and regularization characteristics computed for translating random field and modified tree sequences. We observe that these curves exhibit similar characteristics to that of the translating "greyscale corner" shown in Figure 4.6. The velocity error is a minimum for speeds of 0 and 1 pixel per frame and rises sharply for speeds greater than 1 pixel/frame. As expected the velocity error decreases for the increasing levels of smoothing employed. The superior performance of the first order weighted least squares technique (IF5) is also evident.

Tabulated results of the errors for the five test image sequences are presented in Table 4.2—Table 4.5. In all cases the image sequences where smoothed using a smoothing coefficient of $\alpha = 1.5$ and a weighting of $\lambda = 1.0$ for the augmented second order method. A speed of 0.3 pixels/frame was used for the translating random field and modified tree sequences. The speed is not constant across the field for the other two test sequences, with the speeds varying between 1.73 and 2.26 pixel/frame in the translating tree sequence. In all cases the first order weighted least squares solution (IF5) out performs the similar aperture sized second order solutions (IF6 & IF7). Moreover, the first order weighted least squares solution improves with increasing aperture size at a faster rate than the second order method.

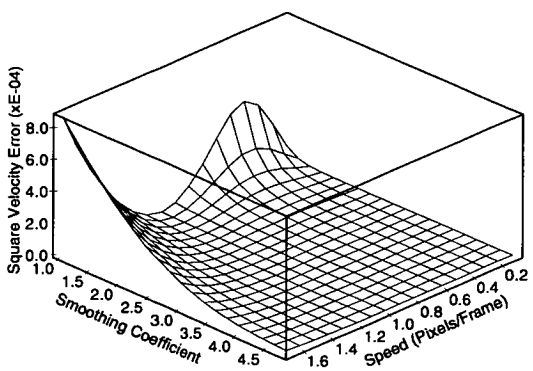
With the diverging optical flow field of the diverging tree sequence, the first order weighted least squares technique and the augmented second order method are nearly equivalent. In



(a)

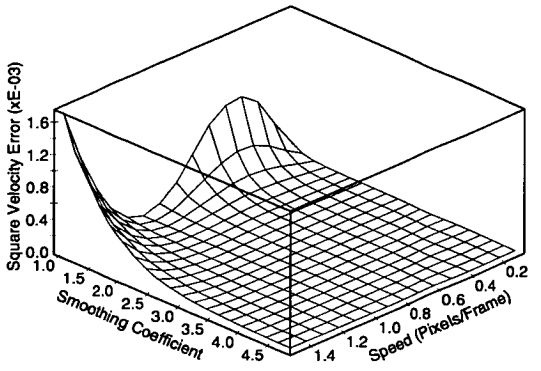


(b)

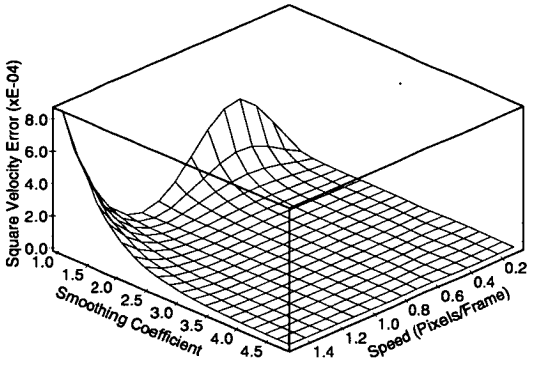


(c)

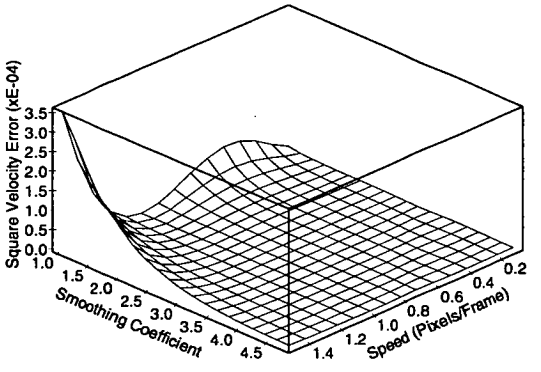
Figure 4.9: *Median square error for the translating random field image sequence. Velocity estimates computed for different speed and smoothing values and methods: (a) IF7, (b) IF6, (c) IF5.*



(a)



(b)



(c)

Figure 4.10: *Median square error for the modified translating tree image sequence. Velocity estimates computed for different speed and smoothing values and methods: (a) IF7, (b) IF6, (c) IF5.*

Method	Aperture Size	Mean	Median	Std. Dev.
IF5	5	7.0048×10^{-3}	1.2041×10^{-3}	0.02×10^{-2}
		2.90°	1.19×10^{-3}°	3.6°
	9	3.1416×10^{-4}	3.4747×10^{-5}	4.14×10^{-3}
		0.40°	3.52×10^{-5}°	0.9°
	13	1.5277×10^{-5}	2.1117×10^{-6}	8.07×10^{-4}
		0.09°	2.13×10^{-6}°	0.2°
	17	4.0405×10^{-7}	2.1383×10^{-7}	6.18×10^{-7}
		0.03°	2.12×10^{-7}°	0.02°
IF6	5	1.2742×10^{-2}	2.0807×10^{-3}	0.02
		4.25°	2.10×10^{-3}°	4.5°
	9	7.7033×10^{-3}	1.7147×10^{-4}	0.02
		2.34°	1.68×10^{-4}°	4.3°
	13	6.9638×10^{-3}	3.0569×10^{-5}	0.02
		1.87°	2.93×10^{-5}°	4.2°
	17	6.7940×10^{-3}	8.3789×10^{-6}	0.02
		1.71°	8.41×10^{-6}°	4.2°
IF7	5	0.1934	3.2609×10^{-3}	3.68
		5.97°	3.25×10^{-3}°	9.8°
	9	0.0393	4.0216×10^{-4}	0.61
		2.74°	3.88×10^{-4}°	6.4°
	13	0.0127	8.4093×10^{-5}	0.23
		1.46°	8.70×10^{-5}°	4.2°
	17	4.1021×10^{-3}	2.5101×10^{-5}	0.08
		0.86°	2.38×10^{-5}°	2.8°

Table 4.2: *Translating random field image sequence. Comparative error results for the three image flow techniques for various aperture sizes.* $\alpha = 1.5$, $\beta = 1.5$, $\lambda = 1.0$.

Method	Aperture Size	Mean	Median	Std. Dev.
IF5	5	2.0181×10^{-3}	3.5645×10^{-4}	2.13×10^{-2}
		$1.41°$	$3.69\times10^{-4}°$	$1.8°$
	9	1.6893×10^{-5}	4.2128×10^{-6}	4.16×10^{-5}
		$0.16°$	$4.12\times10^{-6}°$	$0.2\times10^{-1}°$
	13	5.2394×10^{-7}	8.9175×10^{-8}	1.39×10^{-6}
		$0.03°$	$9.18\times10^{-8}°$	$0.03°$
	17	3.3315×10^{-8}	3.3639×10^{-9}	1.17×10^{-7}
		$5.86\times10^{-3}°$	$3.44\times10^{-9}°$	$7.7\times10^{-3}°$
IF6	5	3.6697×10^{-2}	7.0133×10^{-4}	1.54
		$2.72°$	$7.28\times10^{-4}°$	$5.2°$
	9	6.5656×10^{-4}	3.7308×10^{-5}	7.10×10^{-3}
		$0.62°$	$3.82\times10^{-5}°$	$1.2°$
	13	1.3548×10^{-4}	4.8328×10^{-6}	2.39×10^{-3}
		$0.23°$	$4.99\times10^{-6}°$	$0.6°$
	17	5.8527×10^{-5}	1.1029×10^{-6}	1.76×10^{-3}
		$0.12°$	$1.07\times10^{-6}°$	$0.4°$
IF7	5	3.2437	1.1995×10^{-3}	0.02×10^{2}
		$4.58°$	$1.24\times10^{-3}°$	$9.8°$
	9	0.2770	8.3084×10^{-5}	0.22
		$1.65°$	$7.87\times10^{-5}°$	$5.6°$
	13	0.4222	1.1882×10^{-5}	0.41
		$0.78°$	$1.22\times10^{-5}°$	$3.7°$
	17	8.0041×10^{-3}	2.7049×10^{-6}	0.36
		$0.44°$	$2.66\times10^{-6}°$	$2.6°$

Table 4.3: *Modified translating tree image sequence. Comparative error results of the three image flow techniques for various aperture sizes.* $\alpha = 1.5$, $\beta = 1.5$, $\lambda = 1.0$.

Method	Aperture Size	Mean	Median	Std. Dev.
IF5	5	5.1750×10^{-2}	1.3706×10^{-2}	0.25
		2.62°	1.56×10^{-2}°	3.5°
	9	4.6568×10^{-3}	1.0769×10^{-3}	0.02
		0.78°	8.73×10^{-4}°	0.9°
	13	1.7224×10^{-3}	4.7888×10^{-4}	0.004
		0.50°	1.78×10^{-4}°	0.5°
	17	1.0044×10^{-3}	3.6959×10^{-4}	0.002
		0.40°	5.02×10^{-4}°	0.4°
IF6	5	0.2727	2.0472×10^{-2}	5.21
		4.22°	7.75×10^{-3}°	7.8°
	9	0.1031	3.4841×10^{-3}	2.40
		2.17°	7.33×10^{-3}°	5.4°
	13	0.0770	2.1565×10^{-3}	1.16
		1.91°	5.58×10^{-3}°	5.2°
	17	0.0601	1.9187×10^{-3}	0.73
		1.78°	3.17×10^{-3}°	4.7°
IF7	5	174.1	3.9269×10^{-2}	7923.
		8.82°	5.49×10^{-2}°	19.3°
	9	77.8335	9.7917×10^{-3}	5450.
		6.11°	3.93×10^{-3}°	16.4°
	13	69.3374	6.4871×10^{-3}	4191.
		5.70°	1.03×10^{-2}°	16.2°
	17	90.8489	5.7789×10^{-3}	6249.
		5.42°	5.68×10^{-3}°	15.7°

Table 4.4: *Original translating tree image sequence. Comparative error results of the three image flow techniques for various aperture sizes.* $\alpha = 1.5$, $\beta = 1.5$, $\lambda = 1.0$.

Method	Aperture Size	Mean	Median	Std. Dev.
IF5	5	0.0121	3.2139×10^{-3}	0.04
		3.07°	3.58×10^{-3}°	2.9°
	9	5.5599×10^{-3}	2.0768×10^{-3}	0.01
		2.34°	1.88×10^{-3}°	2.0°
	13	5.2373×10^{-3}	2.2572×10^{-3}	0.01
		2.39°	1.63×10^{-3}°	1.9°
	17	5.7109×10^{-3}	2.5726×10^{-3}	0.01
		2.56°	3.01×10^{-3}°	1.9°
IF6	5	0.0803	5.4825×10^{-3}	0.91
		5.02°	3.97×10^{-3}°	7.2°
	9	0.0714	2.9338×10^{-3}	2.03
		3.98°	2.09×10^{-3}°	6.2°
	13	0.0491	2.6690×10^{-3}	0.94
		3.79°	3.66×10^{-3}°	5.8°
	17	0.0426	2.5610×10^{-3}	0.58
		3.74°	4.69×10^{-3}°	5.7°
IF7	5	52.7167	1.2376×10^{-2}	3.94×10^{3}
		9.44°	8.45×10^{-3}°	15.5°
	9	157.161	7.1373×10^{-3}	1.19×10^{4}
		8.23°	4.00×10^{-3}°	14.9°
	13	43.0388	6.3720×10^{-3}	3.37×10^{3}
		7.95°	9.99×10^{-3}°	14.6°
	17	87.9952	5.9330×10^{-3}	8.49×10^{3}
		7.72°	7.44×10^{-3}°	14.3°

Table 4.5: *Diverging tree image sequence. Comparative error results of the three image flow techniques for various aperture sizes.* $\alpha = 1.5$, $\beta = 1.5$, $\lambda = 1.0$.

this case the assumption of a constant flow field within the aperture is no longer as valid for larger apertures. As a result the error initially decreases for the weighted least squares estimator as gradient calculations improve, but then increases as the window becomes too large and the model become erroneous.

Finally, Figure 4.11 shows the computed image flow field for the moving square sequence (Figure A.1), using the method of first order weighted least squares (IF5). The flow field is now localized around the corners and edges of the square because the algorithm operates only on local image data. This would appear a more practical solution than the smooth image flow generated by Horn and Schunck's technique (IF4) shown in Figure 3.8.

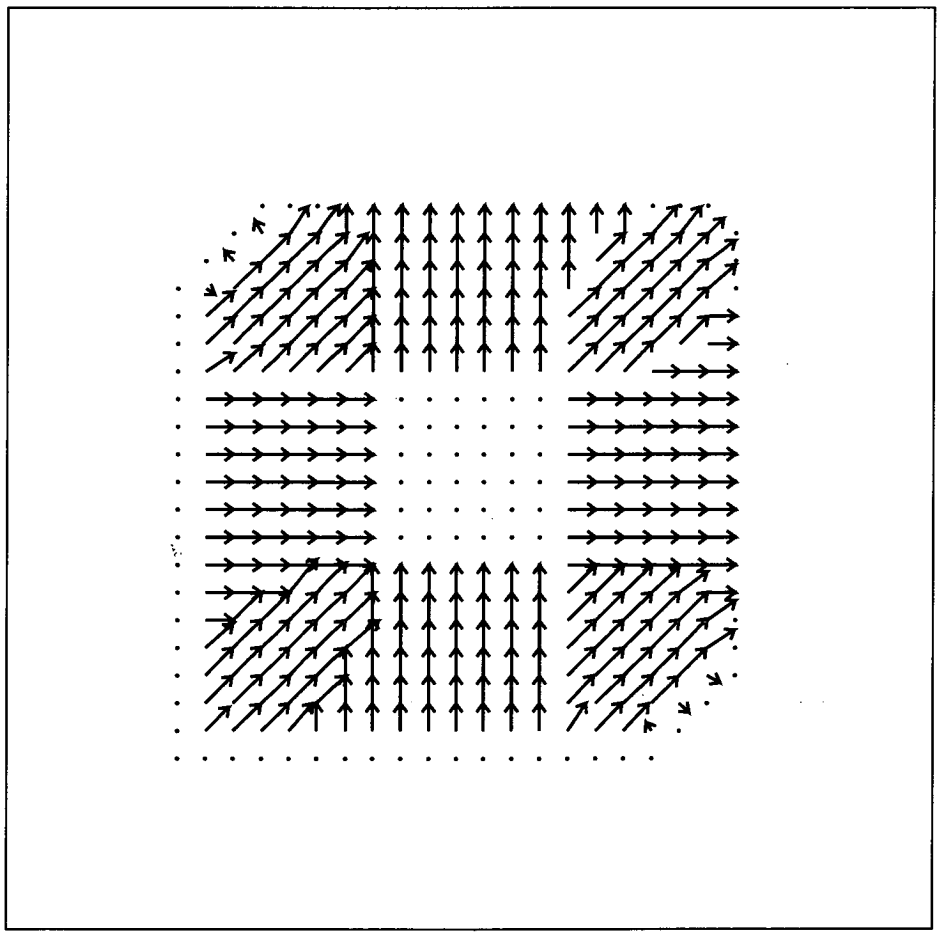


Figure 4.11: *Computed image flow field for the moving square sequence using the first order weighted least squares algorithm (IF5).*

4.8 Measures of Reliability

The use of image flow estimates, on their own, is limited without measures which indicate our confidence in the accuracy or quality of the estimates. Without a measure of reliability we are unable to distinguish between an estimate which is 4% correct to one which is 96% accurate. Moreover, a quantitative measure of the quality of the image flow is important if it is to be used for further processing, for example determining egomotion and depth. It is not, however, possible to directly estimate the velocity error from Eq.(4.16), due to an inability to measure the higher order spatial derivatives, X, or to determine r. Therefore alternative methods of quantifying the reliability of the flow field estimate are required.

There have been a number of methods proposed for estimating the quality of the image flow estimate from the available data. We consider six methods commonly referred to in the literature, listed in Table 4.6, plus a new proposal calculated as the sensitivity of (u, v) to random noise on the image intensity. The performance indicators can be grouped into two general classes: those examining the matrix of spatial gradients $S = A^T W A$ and in particular its conditioning, and those that incorporate the velocity estimate in their measure. The latter group include the temporal derivatives and are therefore dependent on velocity. This is important as both the error function, Eq.(4.16), and our experiments confirm that the expected error is a function of velocity.

Quantitative measures of the correlation between the performance indicators and the flow

Indicator	Description	Speed Dependant
Inverse determinant	$\frac{1}{\Delta(S)}$	·
Inverse minimum eigenvalue	$\frac{1}{\lambda_{min}(S)}$	·
Inverse sum of eigenvalues	$\frac{1}{\sum \lambda(S)}$	·
Sobey's conditioning measure [191]	$\frac{\sum_{i,j} S^2_{[i,j]}}{\Delta S}$	·
Residual (R)	$R = I_x u + I_y v + I_t$	√
Jähne's measure [8]	$\frac{R}{\lambda_{min}(S)}$	√
Sensitivity measure	$\lvert\frac{\partial u}{\partial I}\ \frac{\partial v}{\partial I}\rvert^2$	√

Table 4.6: *Measures of flow field reliability and defining characteristics.*

field error are shown in Table 4.7 — 4.10. These measures were made using the same image sequences analyzed in the previous section. The correlation between the indicator and the error is significant in estimates made by the second order algorithm (IF7), but much less significant for estimates made by the weighted least squares algorithm (IF5). Unfortunately this appears at the expense of the overall accuracy in the estimated image flow, which is significantly worse for IF7 over IF5. This is also apparent in the plots of the inverse minimum eigenvalues against the squared image flow velocity, for a translating random field sequence with velocity $(0.3, 0)$, and the diverging tree sequence. Figure 4.12 and Figure 4.13 show, for each method respectively, a set of graphs displaying log(inverse minimum eigenvalue) against the log(square velocity error) for all of the image flow estimators.

We note that in the second order method the success of the best indicators is attributable to their ability to pick gross flow field errors. These errors occur in regions where the image is bland, and consequently has small derivatives. In this situation the the magnitude of S^{-1} is large, hence amplifying the errors in the spatial and temporal derivatives. Thus, we see a correlation between performance indicators and velocity error. In the weighted least squares approach the effect of errors in the spatial and temporal derivatives is much less, leading to smaller errors in the flow field estimate. A consequence of this is a poorer correlation between the indicator and the actual image flow error.

Investigations into the amount of smoothing employed on the data before computing the gradients were also made. One of the effects of increasing the smoothing is a decrease in the variance of the flow field error and an increase in the observed correlation between indicator and flow field error. Unfortunately, as noted earlier this improvement is at the price of a reduced resolution of the image flow field.

Indicator	Correlation Coefficient		
	IF5	IF6	IF7
Inverse determinant	0.25	0.64	0.68
Inverse minimum eigenvalue	0.21	0.44	0.57
Inverse sum of eigenvalues	0.21	0.33	0.44
Residual	0.19	0.37	0.15
Jähne's measure	0.20	0.21	0.35
Condition number	0.09	0.58	0.62
Sensitivity measure	0.27	0.70	0.70
Gradient Magnitude	0.10	0.15	0.001

Table 4.7: *Translating random field image sequence. Performance measures of flow field indicators, see Table 4.2 for performance parameters and median square error value.*

Indicator	Correlation Coefficient		
	IF5	IF6	IF7
Inverse determinant	0.17	0.26	0.48
Inverse minimum eigenvalue	0.18	0.33	0.45
Inverse sum of eigenvalues	0.14	0.14	0.31
Residual	0.11	0.05	0.35
Jähne's measure	0.18	0.14	0.49
Condition number	0.08	0.33	0.51
Sensitivity measure	0.19	0.37	0.60
Gradient Magnitude	0.12	0.13	0.19

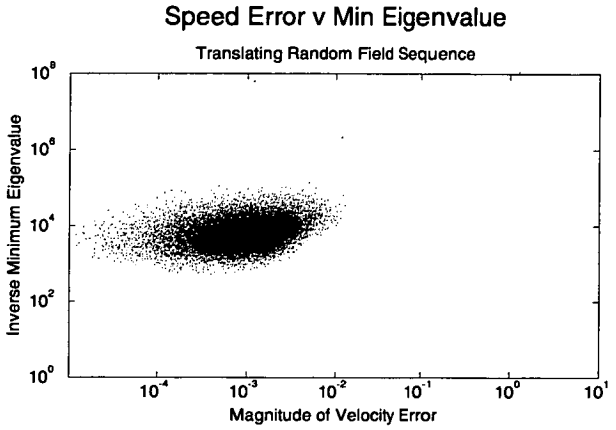
Table 4.8: *Modified translating tree image sequence. Performance measures of flow field indicators, see Table 4.3 for performance parameters and median square error value.*

Indicator	Correlation Coefficient		
	IF5	IF6	IF7
Inverse determinant	0.43	0.55	0.67
Inverse minimum eigenvalue	0.38	0.50	0.58
Inverse sum of eigenvalues	0.37	0.41	0.47
Residual	0.11	0.37	0.49
Jähne's measure	0.24	0.17	0.67
Condition number	0.11	0.30	0.56
Sensitivity measure	0.43	0.60	0.74
Gradient Magnitude	0.34	0.38	0.30

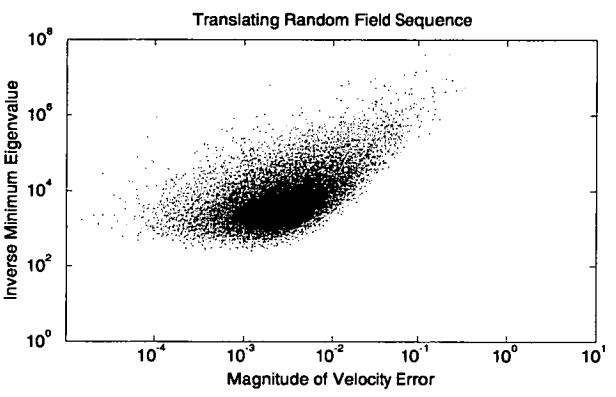
Table 4.9: *Original translating tree image sequence. Performance measures of flow field indicators, see Table 4.4 for performance parameters and median square error value.*

Indicator	Correlation Coefficient		
	IF5	IF6	IF7
Inverse determinant	0.26	0.39	0.55
Inverse minimum eigenvalue	0.32	0.45	0.52
Inverse sum of eigenvalues	0.20	0.23	0.36
Residual	0.02	-0.003	0.52
Jähne's measure	0.19	0.17	0.70
Condition number	0.17	0.40	0.57
Sensitivity measure	0.31	0.52	0.72
Gradient Magnitude	0.31	0.23	0.16

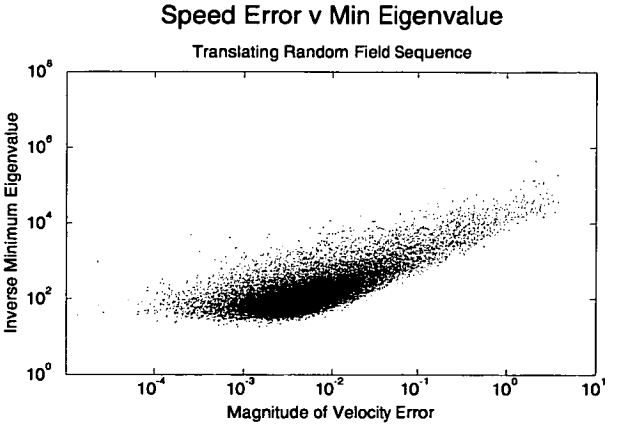
Table 4.10: *Diverging tree image sequence. Performance measures of flow field indicators, see Table 4.5 for performance parameters and median square error value.*



(a)

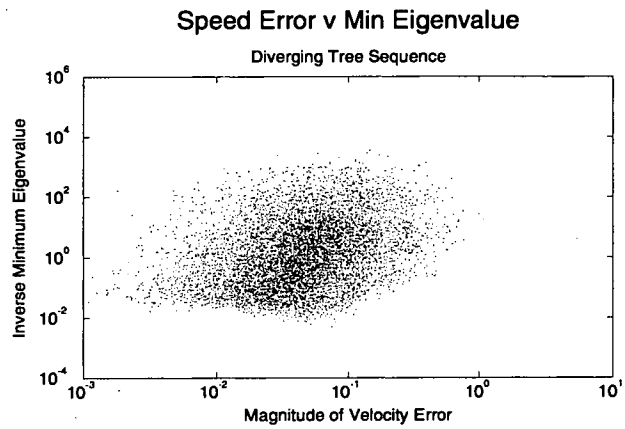


(b)

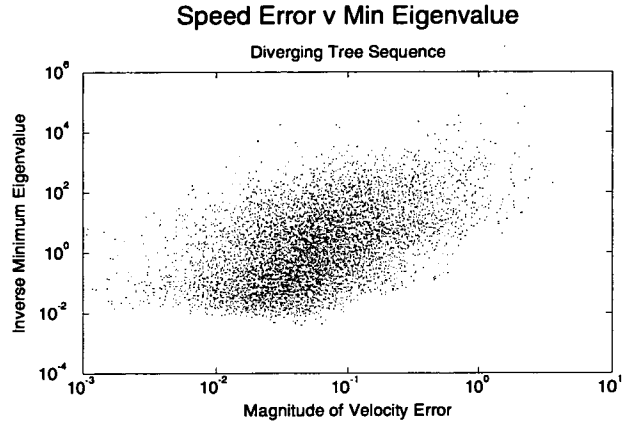


(c)

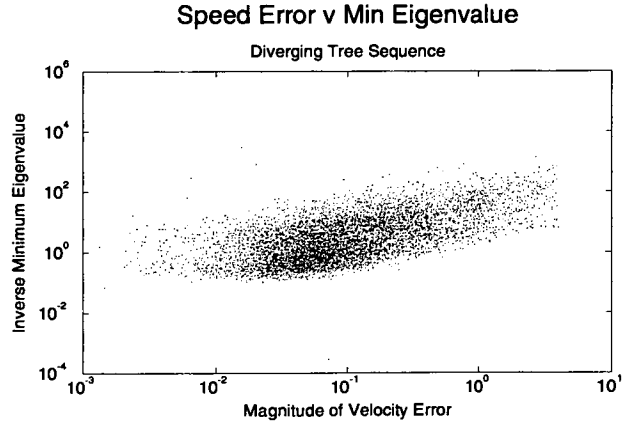
Figure 4.12: *Scatter plot of inverse minimum eigenvalue performance indicator against median square speed errors for the* $(0.3, 0)$ *translating random field sequence, see Figure A.4. Computed using the: (a) IF5, (b) IF6, and (c) IF7 algorithms.*



(a)



(b)



(c)

Figure 4.13: *Scatter plot of inverse minimum eigenvalue performance indicator against median square speed errors for the* $(0.3, 0)$ *diverging tree sequence, see Figure A.6. Computed using the: (a) IF5, (b) IF6, and (c) IF7 algorithms.*

Chapter 5

Colour and Motion

In the previous two chapters a number of techniques for measuring image flow were reviewed, with particular emphasis placed on the differential techniques. Despite the various differences between the algorithms presented, each method is common in that the flow estimate is based solely on processing a greyscale image sequence taken from the sensor. In this chapter an investigation is made into the potential improvements in the velocity estimate obtained through both the addition of a priori knowledge of the camera motion and the extra information contained in colour imagery. The work presented in this chapter is preliminary but ongoing. It has been included in this thesis because of the extensive background mathematical work that has been done. Results obtained, so far, suggest these additional constraints on the image flow do show some promise.

The use of colour imagery to measure image flow is a logical progression from that of greyscale images, given that a large number of "off-the-shelf" CCD cameras are now colour based. An assumption, up until now, has been that the intensity of greyscale images is conserved under motion. Such an assumption holds true only when viewing scenes with Lambertian reflectance properties lit by a diffuse unstructured source. These conditions are rarely, if ever, met in the real world. There is, however, a certain appeal in arguing that while intensity may not be invariant under motion, the chromaticity information from a colour source may be. This chapter therefore, in part, demonstrates the improvements obtainable by such an assumption.

The second part of this chapter investigates the situation where a imaging sensor is mounted

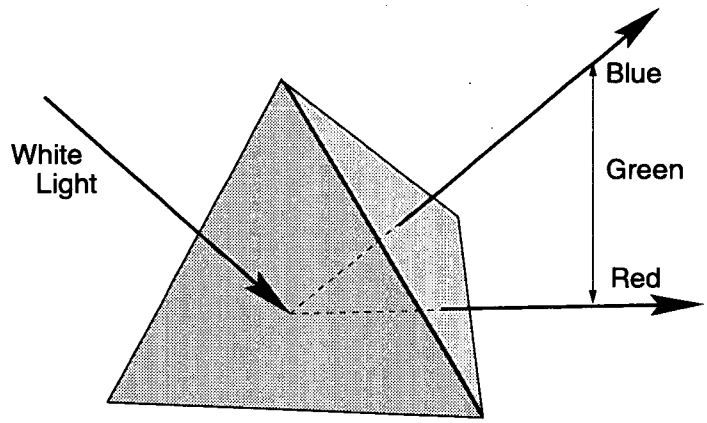
on a movable platform, such as a robotic arm or autonomous robot. Information about the platforms self-motion (ego-motion) is often available and it is shown that this can be used to form a constraint on the image flow.

5.1 Colour Image Formation

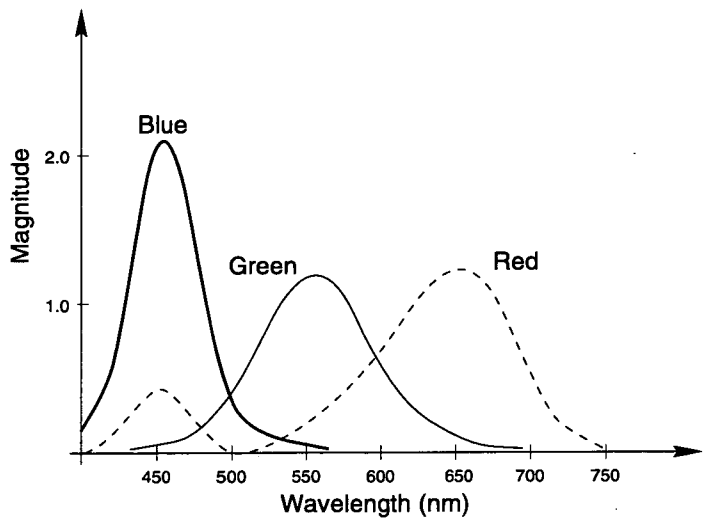
Isaac Newton in 1666 discovered that when a beam of sunlight is passed through a glass prism the emerging beam of light is not white, but consists instead of a continuous spectrum of colours, see Figure 5.1(a). Sunlight, he postulated, consists of a relative balance of all the wavelength of visible light, which when viewed in full as a spectrum shows that no colour in the spectrum ends abruptly but rather blends smoothly into the next. Coloured objects therefore selectively reflect different colours absorbing the other wavelengths. A body that reflects light that is relatively balanced in all wavelengths appears white to the observer. However, a body that favours reflectance in a limited range of the visible spectrum exhibits some shades of colour. In terms of the image formation process, as discussed in Section 2.1, the surface micro-structure in the scene is frequency dependent and the scene BRDF is modeled as a frequency dependent filter.

Colour cameras can be considered to be the mechanical equivalent of the human eye. Colour perception is achieved by first passing the incoming radiation through coloured optic filters before reception by the photo-sensors on the image plane. The coloured optical filters are bandpass filters which select a small section of the electro-magnetic spectrum. Commonly three filters are employed with these being centered in the red, green and blue areas of the visible spectrum Figure 5.1(b). Colour is then described as a variable combination of these three so-called additive primary colours red (R), green (G) and blue (B).

A number of observations can be drawn from the frequency characteristics of the colour sensor. Firstly, no one single frequency can be classified as red, green or blue, as a range of frequencies excite the RGB photo-receptors. However, because no distinction can be discerned between the different frequencies it is possible to describe all reds as just one frequency, 700nm according to the CIE (Commission Internationale de l'Eclairage — the International Commission on Illumination) and similarly for the other two primaries, 435.8nm for blue, and 546.1nm for green. Nevertheless, no combination of three fixed RGB components can generate all the perceivable spectral colours [5].



(a)



(b)

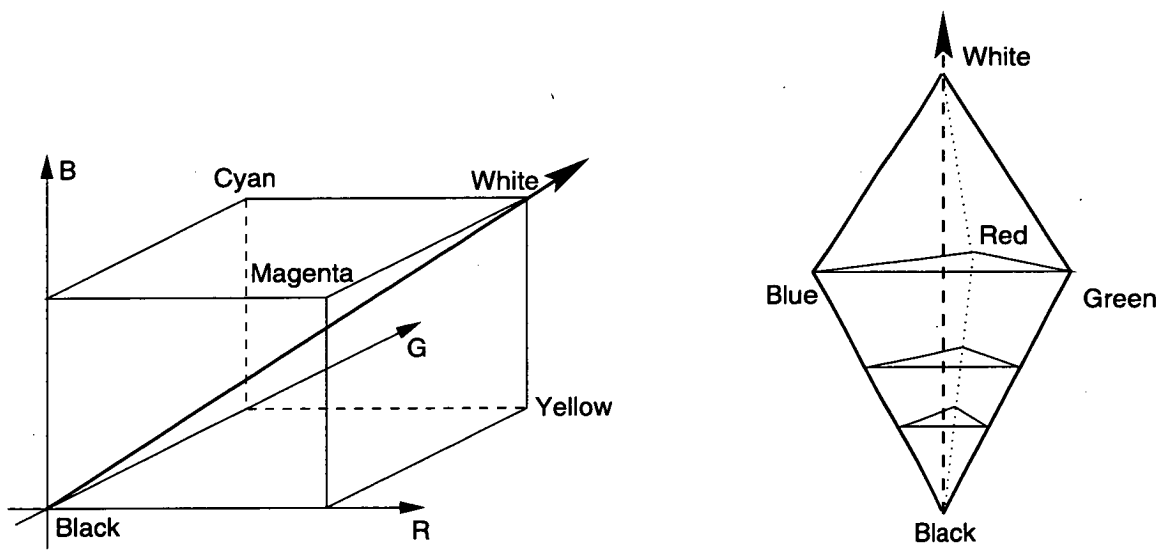
Figure 5.1: *(a) White light is made up of a relative balance of a spectrum of colours. (b) Spectral response distribution of the "standard" observer to the three additive primary colours.*

5.2 Colour Model Transforms

The purpose of a colour model is to facilitate the specification of colours in a standard, generally acceptable way. In essence, a colour model is a specification of a three-dimensional coordinate system and a subspace within that system where each colour is represented by a single point. Most colour models in use today are oriented either towards physical devices (such as colour monitors, printers, cameras) or towards applications where data manipulation is a goal (TV broadcasting, compression, enhancement, machine vision). The most common models used in practice are: the RGB model for monitors and a broad class of video-cameras; the CMYK model for colour printers (based on subtractive or colourant primaries); the YIQ model, which is the standard for broadcast colour TV; and the HSI (HSV) model used for computer vision. In addition a number of other models are also widely used.

In the RGB model, each colour appears in its primary spectral components of red, green and blue. This model is based on a Cartesian coordinate system or colour cube as shown in Figure 5.2(a) in which the primaries are at diagonally opposite corners; the secondaries at the interceding corners; black is at the origin; with white the furtherest opposite corner. In this model, the greyscale extends from black to white along the line joining these two points. Colours are points on or inside the cube and are defined by vectors extending from the origin. These vectors are the tristimulus RGB values. For convenience the digital world assumes that each RGB values lies in the range [0, 255] (8 bits), thus giving a 24-bit colour model.

The characteristics generally used to distinguish one colour from another are brightness, hue and saturation. Brightness is a subjective descriptor that embodies the achromatic notion of intensity. In the machine vision problem luminance, or intensity, are the measures used to indicate the amount of energy an observer perceives from a light source. This is equivalent to the greyscale value. Hue is an attribute associated with the dominant wavelength in a mixture of light waves. Thus hue represents the dominant colour perceived by an observer; when we call an object red, orange or yellow we are specifying its hue. Saturation refers to relative purity or the amount of white light mixed with a hue. The pure spectrum colours are fully saturated. Colours such as pink (red and white) and lavender (violet and white) are less saturated, with the degree of saturation being inversely proportional to the amount of white light added. Hue and saturation taken together are called chromaticity, and therefore, a colour may be characterized by its intensity and chromaticity.



(a) (b)

Figure 5.2: *Colour models: (a) The tristimulus RGB colour cube, (b) The HSI colour solid.*

Hue, saturation and intensity altogether describe the HSI colour model. It owes its usefulness to two principal facts. Firstly, chromaticity is intimately related to the way in which humans perceive colour [74, 84, 95, 208]. Secondly, the intensity component, I, is decoupled from the chromaticity in the image, allowing greyscale image processing without damaging the apparent colour. This is inherently useful because the chromaticity components of this model should be largely invariant of both lighting variation and shadows.

The RGB to HSI colour model transform, involves a transformation from the tristimulus RGB colour cube to the HSI colour solid, as shown in Figure 5.2(b). The HSI colour solid is constructed by taking parallel slices through the RGB colour cube perpendicular to and along the black-white line. The slices are then stacked on top of each other in order from black to white. Each slice is an equilateral triangle with the edges being the fully saturated colours; the interior of the triangle therefore represents colours which are not fully saturated. RGB colour values can be converted to HSI values by the following conversion rules [5],

$$I = \tfrac{1}{3}(R+G+B), \tag{5.1a}$$

$$S = 1 - \frac{3}{I}\min\{R,G,B\}, \tag{5.1b}$$

$$H = \begin{cases} \cos^{-1}\left(\dfrac{R-\frac{1}{2}(G+B)}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right) & \text{if } G > B \\ -\cos^{-1}\left(\dfrac{R-\frac{1}{2}(G+B)}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right) & \text{otherwise} \end{cases} \tag{5.1c}$$

Hue is thus measured as an angle and lies in the interval $[-\pi, \pi)$. For a hue of pure red $H = 0$; pure green $H = \frac{2}{3}\pi$, and pure blue $H = -\frac{2}{3}\pi$.

5.3 Colour in Motion

The usual image quantity which is assumed to be conserved under motion is the light intensity. This leads to the usual under-determined problem of,

$$\frac{dI}{dt} = I_t + I_x u + I_y v = 0, \tag{3.11}$$

for which additional constraints are required to form a solution. Traditionally this has been achieved by applying some smoothness constraint on the velocity field, or a constraint

on the gradient of the image intensity. The use of colour imagery, with its increase in potentially useful information, may overcome some of the limitations of the earlier models. Providing the intensity and chromaticity do not vary in exactly the same way, then either less smoothing is required in the case of a global method, Section 3.3, or a smaller aperture size, Section 4.1, may be used in the case of a local method. This is highly desirable because it enables a greater resolution of the image flow field to be computed.

As colour provides three bands of information, the intensity conservation constraint equation at a point for each of the HSI bands gives,

$$E^H(x,y) = (I_x^H u + I_y^H v + I_t^H)^2, \tag{5.2a}$$

$$E^S(x,y) = (I_x^S u + I_y^S v + I_t^S)^2, \tag{5.2b}$$

$$E^I(x,y) = (I_x^I u + I_y^I v + I_t^I)^2. \tag{5.2c}$$

The error contributed by the deviation, $E^H(x,y)$, $E^S(x,y)$, $E^I(x,y)$, of each of the three bands gives rise to the Lagrange error function,

$$E^t(x,y) = \epsilon_1^2 E^H(x,y) + \epsilon_2^2 E^S(x,y) + E^I(x,y) + \gamma^2 E^{sH}(x,y), \tag{5.2d}$$

where E^{sH} is Horn and Schunck's smoothness constraint, Eq.(3.13c). Different weightings may be assigned to the hue, saturation and smoothness constraint through $\epsilon_1, \epsilon_2, \gamma$. Minimization of Eq.(5.2d) with respect to (u,v) gives,

$$[u^{\tau+1}\ v^{\tau+1}]^T = (A^T W A + \gamma^2 I)^{-1}(\gamma^2[\bar{u}^\tau\ \bar{v}^\tau]^T + A^T W b), \tag{5.3}$$

where

$$A = \begin{bmatrix} I_x^H & I_y^H \\ I_x^S & I_y^S \\ I_x^I & I_y^I \end{bmatrix} \quad \text{and,} \quad b = \begin{bmatrix} I_t^H \\ I_t^S \\ I_t^I \end{bmatrix} \quad \text{and,} \quad W = \begin{bmatrix} \epsilon_1^2 & 0 & 0 \\ 0 & \epsilon_2^2 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Equation (5.3) is solved using an iterative Gauss-Seidel technique with the initial condition ($\tau = 0$) of $(\bar{u}, \bar{v}) = (0,0)$. This is a global differential method and can be thought of as a "colourized" Horn and Schunck method. It is denoted as method IF8.

As in the case of greyscale imagery smoothing of the gradient terms is required, with this process normally being performed prior to the differencing computations. For conformity the same smoothing method is applied to the colour image sequences. Smoothing, however,

is applied to the HSI data and not the RGB, so as to maintain colour consistency [5]. Note that special care must be taken with both the smoothing and differential computations involving the hue measurements as it is a discontinuous function.

Instead of using a global smoothness constraint the errors in Eq.(5.2) can also be minimized over a localized aperture rather than the entire image. Using an underlying assumption of a constant velocity over the aperture, a "colourized" first order weighted least squares technique solves,

$$\begin{aligned}\min_{u,v} &\iint \zeta_1^2 W(x-x_0, y-y_0)\left(I_x^H u + I_y^H v + I_t^H\right)^2 dx\,dy, \\ &+ \iint \zeta_2^2 W(x-x_0, y-y_0)\left(I_x^S u + I_y^S v + I_t^S\right)^2 dx\,dy, \\ &+ \iint \zeta_3^2 W(x-x_0, y-y_0)\left(I_x^I u + I_y^I v + I_t^I\right)^2 dx\,dy, \end{aligned} \tag{5.4}$$

where $W(x,y)$ denotes a symmetric window function of finite size and $\zeta_1, \zeta_2, \zeta_3$ are weights applied to differentiate between the relative important of the three colour bands.

Rearranged Eq.(5.4) to form a set of equations produces,

$$A^T W^\zeta A \begin{bmatrix} u \\ v \end{bmatrix} = A^T W^\zeta b. \tag{5.5}$$

Here the spatial and temporal gradient measurements are arranged as,

$$A = \begin{bmatrix} I_x^H(x_1,y_1) & I_y^H(x_1,y_1) \\ I_x^H(x_2,y_2) & I_y^H(x_2,y_2) \\ \vdots & \vdots \\ I_x^H(x_n,y_n) & I_y^H(x_n,y_n) \\ I_x^S(x_1,y_1) & I_y^S(x_1,y_1) \\ I_x^S(x_2,y_2) & I_y^S(x_2,y_2) \\ \vdots & \vdots \\ I_x^S(x_n,y_n) & I_y^S(x_n,y_n) \\ I_x^I(x_1,y_1) & I_y^I(x_1,y_1) \\ I_x^I(x_2,y_2) & I_y^I(x_2,y_2) \\ \vdots & \vdots \\ I_x^I(x_n,y_n) & I_y^I(x_n,y_n) \end{bmatrix}, \quad \text{and,} \quad b = \begin{bmatrix} -I_t^H(x_1,y_1) \\ -I_t^H(x_2,y_2) \\ \vdots \\ -I_t^H(x_n,y_n) \\ -I_t^S(x_1,y_1) \\ -I_t^S(x_2,y_2) \\ \vdots \\ -I_t^S(x_n,y_n) \\ -I_t^I(x_1,y_1) \\ -I_t^I(x_2,y_2) \\ \vdots \\ -I_t^I(x_n,y_n) \end{bmatrix}.$$

The weights for each of the colour bands can be incorporated with W to form a single weight matrix, W^ζ, which places different significance on each gradient measure depending upon the

data band (hue, saturation, or intensity) as well as their location within the neighbourhood. A significant change in the gradients over the aperture then means a solution is computable. As with the greyscale algorithms a significant advantage is expected from this local method over the global method due to its localization of errors, Section 4.1. This method is referred to as IF9.

5.4 Colour Performance

The motivation for this work stems from recent work by researchers, particularly in the psychophysical field, who have shown that colour may also play a significant part in the discrimination of image flow in situations where intensity information is either minimal, or gives rise to contradictions in the perceived motion [74, 84, 95, 208]. Two different scenarios can be formed to test this conjecture, namely: the isoluminance problem and the problem of shadows.

When the intensity information is minimal, for example in the relative motion of two structures of low intensity difference, motion can be estimated, albeit inaccurately, by use of the much stronger chromatic variation. This is referred to this as the isoluminance problem [208].

The relative motion of a shadow in a region is an example of a contradictory stimulus, since the changing intensity field gives rise to an apparent motion. In this case chromaticity fields are relatively immune to the apparent motion, [74, 95]. Cancellation of the apparent motion would eliminate the Lambertian reflectance assumption implicit in the conservation equation for image intensity. This is a significant and often common problem in machine vision tasks, as moving light sources will give rise to shifting shadows.

Figure A.8, shows an example of the "strips" colour test sequence. This sequence is based on a random field sequence where the intensity function of each image was generated using a Fourier phase shifted random field. The hue function alternates in vertical strips of red and green while the saturation function is held at full saturation. The intensity translates in the vertical direction at a sub-pixel rate, while hue translates horizontally at 1 pixel/frame. No variation in the intensity function is observed in the horizontal direction, consequently both intensity based methods IF4 and IF5 are only able to estimate the vertical components of the optical flow, Figure 5.3(a) & (b). Methods IF8 and IF9, however, produce a more

acceptable result for this translating isoluminance example, Figure 5.3(c) & (d). Although the colour methods do provide a good indication of the true direction of motion, in general, the accuracy of the estimates are still poor.

A second example, Figure A.10, shows a synthetically generated test sequence using a planar translating camera model. The scene consists of a number of objects with sharp well formed edges and large areas of inactivity. As expected global method IF4 implemented with a significant smoothing weight factor produced an accurate estimate for the majority of the field of view, see Figure 5.4(a). In particular it should be noted that the spread of the velocity from points of high gradient information to areas of small gradient has been limited by restricting the number of iterations. Even with a very small window size of 3×3, localized method IF9 provides an estimate more in keeping with expectations, Figure 5.4(b). In addition it is more quickly computed. Clearly, therefore, many of the advantages of the localized differential methods over global methods for greyscale images also hold for colour images, a result which is not unexpected.

The third example, Figure A.11, shows a sequence which was captured with poor contrast. We observe a number of errors in the flow estimated using Horn and Schunck's IF4 algorithm. These are to be expected from consideration of the aperture problem, see Figure 5.5(a). The errors can be ameliorated if much greater smoothing is enforced, which is effectively achieved by using a larger number of iterations. Furthermore, poor intensity gradient measures in the bland intensity region, especially in the bottom left quadrant of the image produce a nonuniform velocity field. A high level of smoothing in this case would result in corruption of large sections of the field. Results more in keeping with the expected optical flow are estimated by methods IF5, IF8 and IF9 respectively. With the "colourized" Horn and Schunck method IF8, Figure 5.5(b), substantially less smoothing is required with the use of chromaticity information. The additional chromaticity information also provides an extra degree of stability to the first-order weighted least square differential technique (IF9).

Figure A.9, shows three frames from a simulated image sequence in which we tested the performance of algorithms IF5 and IF9 in the case of isoluminance and shadow-motion. As can be seen a shadow bar is moving left to right across an upwardly translating isoluminant background. Figure 5.6(a) shows that method IF5 measures an apparent motion due to the moving shadow, but is unable to detect the true background motion. Method IF9

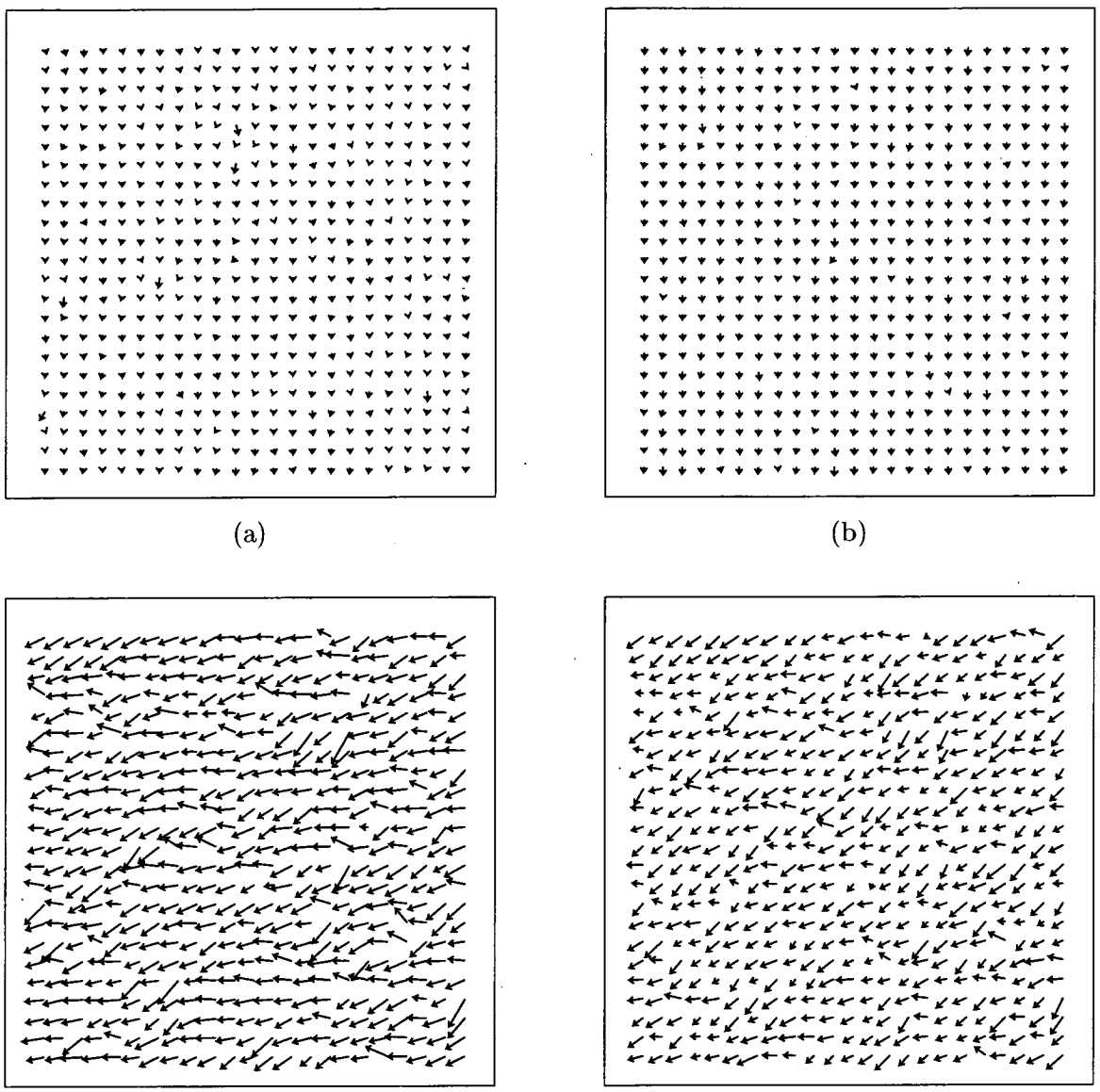
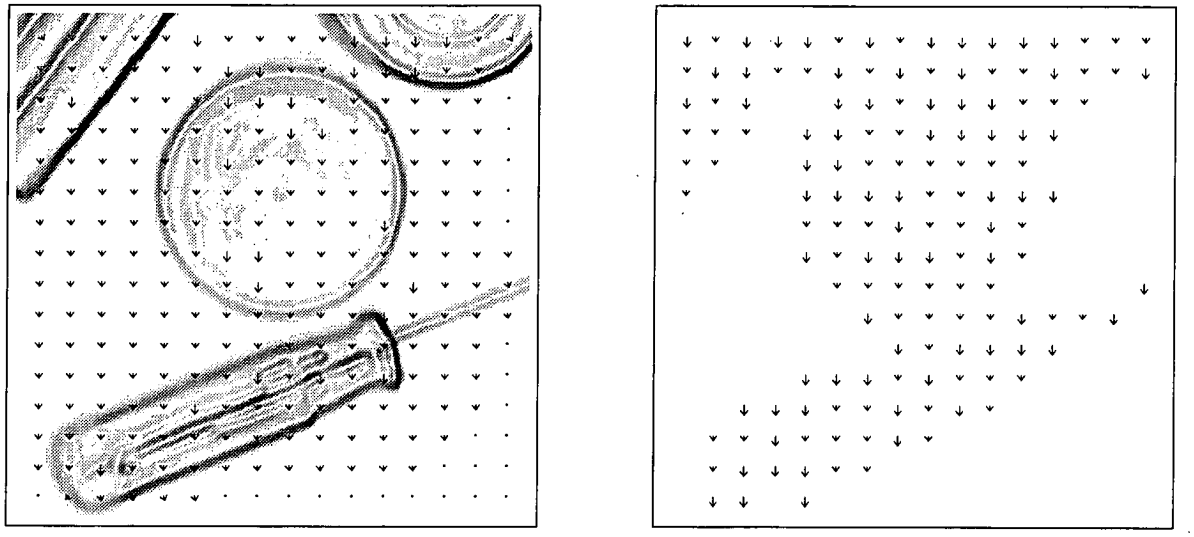
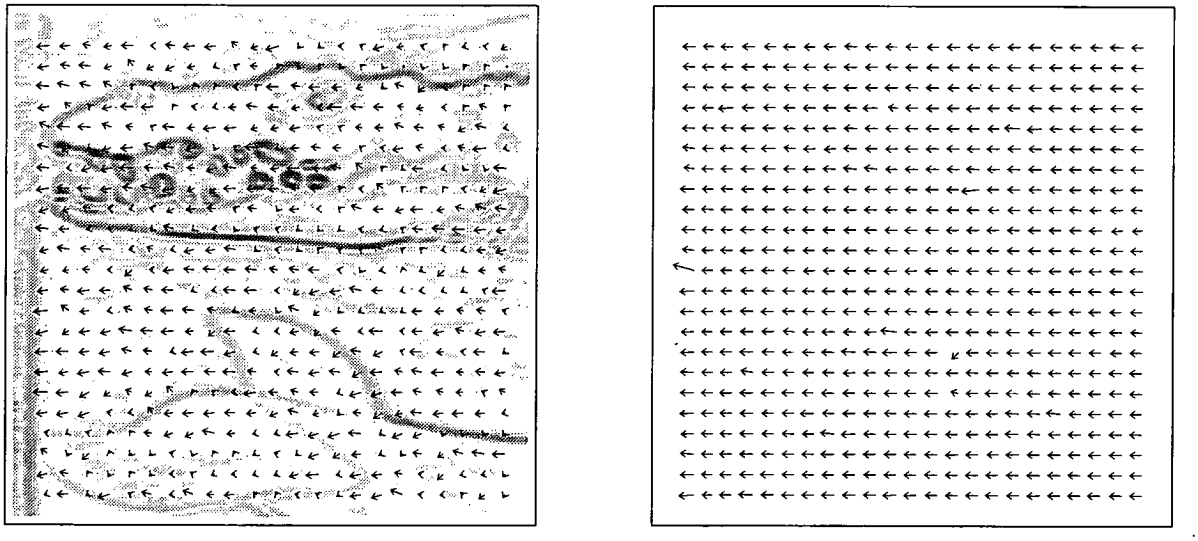


Figure 5.3: *Computed image flow from sequence shown in Figure A.8 using methods: (a) IF4, (b) IF5, (c) IF8 and (d) IF9. In each example a spatiotemporal smoothing coefficient of* $\alpha = 3.0$ *was used.*



(a) (b)

Figure 5.4: *Computed image flow from sequence shown in Figure A.10 using methods: (a) IF4 (1000 iterations) and (b) IF9. In each example a spatiotemporal smoothing coefficient of $\alpha = 3.0$ was used.*



(a) (b)

Figure 5.5: *Computed image flow from sequence shown in Figure A.11 using methods: (a) IF4 and (b) IF8 for $\gamma = 5$ and 100 iterations. In each example a spatiotemporal smoothing coefficient of $\alpha = 3.0$ was used.*

on the same sequence accurately measures the translating background and perceives some motion due to the shadow, see Figure 5.6(b). This result is not unexpected since a least-squares solution attempts only to minimize the error in an inconsistent over-determined system. The effect of the shadow can be ameliorated by giving a small weighting to the intensity components, whereupon the motion is estimated only from the hue and saturation. The errors that are present in the background motion are due to the rapid changes in the saturation gradient field which swamps the other gradients in the intensity and hue fields.

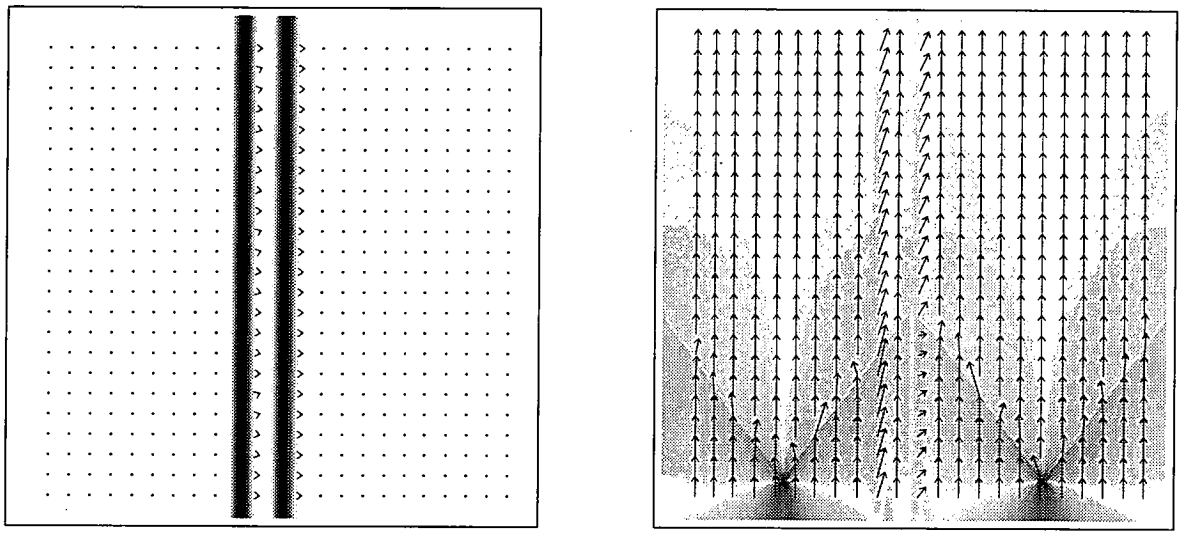
5.5 General Camera Motion and Optical Flow

Optical and image flow arise from either motion within the scene, and, or because of movement of the sensor. If the image flow is due to the movement of the sensor, such as in the case of a camera mounted on a robotic arm, then often the physical motion parameters are available. This is because they are usually required in the motion control process. The physical motion parameters of the sensor can then be used as a constraint upon the image flow estimation process, as is developed in the following discussion. The addition of another independent constraint should, hopefully, increase robustness of the algorithm, as well as increasing the permissible flow field resolution.

As an example, consider a coordinate system at the vanishing point or centre of the camera lens at time t_0. By convention we model the camera motion to be acting at the camera's focal or vanishing point. We could equally, however, have used the camera's principal point (where the optical axis meets the image plane). At time t_1 the camera is situated at a new location after undergoing some general movement. The motion of the camera is defined as the sum of two different terms, a linear translation T and a rotation Q. The translational displacement vector is defined,

$$T = (U\Delta t, V\Delta t, W\Delta t), \tag{5.6}$$

where (U, V, W) are the three-dimensional translation velocity components and a time interval, $\Delta t = t_1 - t_0$, usually equal to the mean time between captured frames. The rotational



(a) (b)

Figure 5.6: *Computed image flow from sequence shown in Figure A.9 using methods: (a) IF5 and (b) IF9.*

motion is defined by the 4 term "rotation vector" or quaternion,

$$Q_{\omega\Delta t,\vec{N}} = \begin{bmatrix} \cos(\omega\Delta t\theta/2) \\ N_1 \sin(\omega\Delta t/2) \\ N_2 \sin(\omega\Delta t/2) \\ N_3 \sin(\omega\Delta t/2) \end{bmatrix} \tag{5.7a}$$

,

where ω is the rate of rotation about the axis $\vec{N}$. The rotation operator is defined as,

$$\begin{aligned} (X_Q, Y_Q, Z_Q) &= \text{Rotation}\{(X, Y, Z)\} \\ &= Q_{\theta,\vec{N}} \otimes (0, X, Y, Z) \otimes Q^*_{\theta,\vec{N}}, \end{aligned} \tag{5.7b}$$

where $\otimes$ is the quaternion multiplication operator and Q^* is the quaternion conjugate. (X, Y, Z) and (X_Q, Y_Q, Z_Q) are the initial and final positions of the object after rotation respectively.

The quaternion is an alternative transformation to the rotation matrix and it is traditionally employed in robotics. It has the some effect of transforming points in an input space to the output space by means of a rotation about a rotation vector. A review of both rotation matrices and quaternion algebra is presented in Appendix B. The advantages of the quaternion representation lie in its compact definition of 4 terms as opposed to the 9 terms required by the 3×3 rotation matrix. Thus, only one constraint equation is required to define the 3 independent rotation terms: pitch, roll and yaw. Moreover, the constraint is linear, and hence does not involve trigonometric terms. For this reason we find the quaternion representation superior for solving least squares problems.

Surface points in the scene move relative (but in the opposite direction and sense) to camera motion. The new position (X_2, Y_2, Z_2) of a scene point relative to the original camera coordinate system is given by,

$$(X_2, Y_2, Z_2) = Q_{\omega\Delta t,\vec{N}} \otimes (0, X_1, Y_1, Z_1) \otimes Q^*_{\omega\Delta t,\vec{N}} - T. \tag{5.8}$$

The instantaneous camera motion at the point (X_1, Y_1, Z_1) is given by,

$$\begin{aligned}
\begin{bmatrix} X_1' \\ Y_1' \\ Z_1' \end{bmatrix} &= \lim_{\Delta t \to 0} \frac{(X_2, Y_2, Z_2) - (X_1, Y_1, Z_1)}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{Q_{\omega \Delta t, \vec{N}} \otimes (0, X_1, Y_1, Z_1) \otimes Q^*_{\omega \Delta t, \vec{N}} - T - (X_1, Y_1, Z_1)}{\Delta t} \\
&= \begin{bmatrix} Y_1 \Omega_3 - Z_1 \Omega_2 - U \\ Z_1 \Omega_1 - X_1 \Omega_3 - V \\ X_1 \Omega_2 - Y_1 \Omega_1 - W \end{bmatrix} \\
&= [X_1\ Y_1\ Z_1]^T \times \Omega - T, \qquad (5.9)
\end{aligned}$$

where $\Omega = [\Omega_1\ \Omega_2\ \Omega_3]^T = \vec{N} \sin(\omega \Delta t)$.

The problem can be further complicated by relaxing the camera model and allowing the camera's focal length to also change. In such a situation the image plane remains stationary while the lens is displaced. The displacement is thus a change in focal position giving rise not only to a scaling of the image but also a small translation along the optical axis, Z direction. Rewriting the translation component as,

$$T = (U\Delta t, V\Delta t, W\Delta t + \Lambda \Delta t), \qquad (5.10)$$

where $\Lambda \Delta t$ is the change in focal length. The new focal length for the perspective transform is,

$$f_2 = f_1 + \Lambda \Delta t. \qquad (5.11)$$

The three-dimensional scene points are projected onto the image plane through the perspective transform,

$$\begin{aligned}
(x, y) &= P\{(X, Y, Z), f\} \\
&= \left(\frac{Xf}{Z}, \frac{Yf}{Z}\right). \qquad (5.12)
\end{aligned}$$

Therefore at time t_1 the point at (X_1, Y_1, Z_1) intersects the image plane at,

$$(x_1, y_1) = P\{(X_1, Y_1, Z_1), f_1\}, \qquad (5.13)$$

and at time t_2 after the camera, and therefore the coordinate system, has moved the same world point will intersect the image plane at,

$$(x_2, y_2) = P\left\{Q_{\omega\Delta t,\vec{N}} \otimes (0, X_1, Y_1, Z_1) \otimes Q^*_{\omega\Delta t,\vec{N}} - T, f_1 + \Lambda\Delta t\right\}. \tag{5.14}$$

The optical flow at a point is defined as the instantaneous rate of change of the imaged scene,

$$\begin{aligned}
\begin{bmatrix} x'_1 \\ y'_1 \end{bmatrix} &= \lim_{\Delta t\to 0} \frac{(x_2, y_2) - (x_1, y_1)}{\Delta t} \\
&= \lim_{\Delta t\to 0} \frac{P\left\{Q_{\omega\Delta t,\vec{N}} \otimes (0, X_1, Y_1, Z_1) \otimes Q^*_{\omega\Delta t,\vec{N}} - T, f_1 + \Lambda\Delta t\right\} - P\{(X_1, Y_1, Z_1), f_1\}}{\Delta t} \\
&= \begin{bmatrix} \frac{1}{Z}(x_1 W + x_1\Lambda - f_1 U) + y_1\Omega_3 - f_1\Omega_2 - \frac{x_1}{f_1}(x_1\Omega_2 - y_1\Omega_1) + x_1\frac{\Lambda}{f_1} \\ \frac{1}{Z}(y_1 W + y_1\Lambda - f_1 V) + f_1\Omega_1 - x_1\Omega_3 - \frac{y_1}{f_1}(x_1\Omega_2 - y_1\Omega_1) + y_1\frac{\Lambda}{f_1} \end{bmatrix}.
\end{aligned} \tag{5.15}$$

This expression has been shown by many authors for the case $\Lambda = 0$, [7,8,68]. In general, the literature has dealt with the case of $\Lambda \neq 0$ solely in the context of obtaining depth from focus [86,170].

In practice only average displacement rates can be measured, because frames can neither be captured nor computations made in infinitely small intervals. For a normalized frame rate, and dropping the time subscript, the displacement rate optical flow is given by,

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \dfrac{Z\begin{pmatrix} (q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)\Lambda + 2(q_2{}^2 - q_4{}^2) \\ (f+\Lambda) - 2(q_1q_3 + q_2q_4)x \\ +2(q_2q_3 + q_1q_4)(f+\Lambda)y \\ +2(q_1q_2 - q_3q_4)xy + 2(q_2q_4 - q_1q_3)f(f+\Lambda) \end{pmatrix} x - (f+\Lambda)fU + xfW + xf\Lambda}{Z((q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)f + 2(q_2q_4 + q_1q_3)x + 2(q_3q_4 - q_1q_2)y) - fW - f\Lambda} \\ \dfrac{Z\begin{pmatrix} 2(f+\Lambda)(q_2q_3 - q_1q_4)x \\ +\begin{pmatrix} (q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)\Lambda + 2(q_3{}^2 - q_4{}^2) \\ (f+\Lambda) + 2(q_1q_2 + q_3q_4)y \end{pmatrix} y \\ -2(q_1q_3 + q_2q_4)xy + 2(q_3q_4 + q_1q_2)f(f+\Lambda) \end{pmatrix} - (f+\Lambda)fV + yfW + yf\Lambda}{Z((q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)f + 2(q_2q_4 + q_1q_3)x + 2(q_3q_4 - q_1q_2)y) - fW - f\Lambda} \end{bmatrix}. \tag{5.16a}$$

This unwieldy expression can be simplified into the form [1] ,

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{ZA_R - K_T + xM_T}{ZD_R - M_T} \\ \frac{ZB_R - L_T + yM_T}{ZD_R - M_T} \end{bmatrix}, \tag{5.16b}$$

where,

$$A_R = \begin{pmatrix} (q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)\Lambda \\ +2(q_2{}^2 - q_4{}^2)(f+\Lambda) \\ -2(q_1q_3 + q_2q_4)x \end{pmatrix} x + 2(q_2q_3 + q_1q_4)(f+\Lambda)y \quad , \tag{5.16c}$$

$$B_R = \begin{array}{l} +2(q_1q_2 - q_3q_4)xy + 2(q_2q_4 - q_1q_3)f(f+\Lambda) \\ 2(f+\Lambda)(q_2q_3 - q_1q_4)x + \begin{pmatrix} (q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)\Lambda \\ +2(q_3{}^2 - q_4{}^2)(f+\Lambda) \\ +2(q_1q_2 - q_3q_4)y \end{pmatrix} y \end{array} \quad , \tag{5.16d}$$

$$\begin{aligned} &-2(q_1q_3 + q_2q_4)xy + 2(q_3q_4 + q_1q_2)f(f+\Lambda) \\ D_R &= (q_1{}^2 - q_2{}^2 - q_3{}^2 + q_4{}^2)f + 2(q_2q_4 + q_1q_3)x + 2(q_3q_4 - q_1q_2)y, \end{aligned} \tag{5.16e}$$

$$K_T = (f+\Lambda)fU, \tag{5.16f}$$

$$L_T = (f+\Lambda)fV, \tag{5.16g}$$

$$M_T = f(W+\Lambda). \tag{5.16h}$$

5.6 An Enhanced Image Flow Technique

We now show that by combining the two equations of Eq.(5.16b) a relationship independent of depth Z can be formed. First, manipulate the equations into terms involving Z on the left-hand side,

$$\begin{bmatrix} ZD_Ru - M_Tu \\ ZD_Rv - M_Tv \end{bmatrix} = \begin{bmatrix} ZA_R - K_T + M_Tx \\ ZB_R - L_T + M_Ty \end{bmatrix}$$

$$\Leftrightarrow \quad \begin{bmatrix} Z(D_Ru - A_R) \\ Z(D_Rv - B_R) \end{bmatrix} = \begin{bmatrix} M_Tu - K_T + M_Tx \\ M_Tv - L_T + M_Ty \end{bmatrix}.$$

[1] This algebraic work was performed using the commercial symbolic mathematical packages *Mathematica* and *Maple*.

Combining using division,

$$\frac{D_R v - B_R}{D_R u - A_R} = \frac{M_T v - L_T + M_T y}{M_T u - K_T + M_T x}$$

$$\Leftrightarrow \quad (D_R v - B_R)(M_T u - K_T + M_T x) = (D_R u - A_R)(M_T v - L_T + M_T y)$$

$$\Leftrightarrow \quad (D_R M_T x - D_R K_T + A_R M_T)v = (D_R M_T y - D_R L_T + B_R M_T)u$$
$$+A_R L_T - A_R M_T y - B_R K_T + B_R M_T x$$

$$\Leftrightarrow \quad v = \frac{(yM_T - L_T)D_R + B_R M_T}{(xM_T - K_T)D_R + A_R M_T}u + \frac{(xM_T - K_T)B_R - (yM_T - L_T)A_R}{(xM_T - K_T)D_R + A_R M_T}. \tag{5.17}$$

This equation, Eq.(5.17), is particularly important as it shows that the optical flow displacement rate is constrained in a manner which is only dependent on the motion parameters and is independent of the scene.

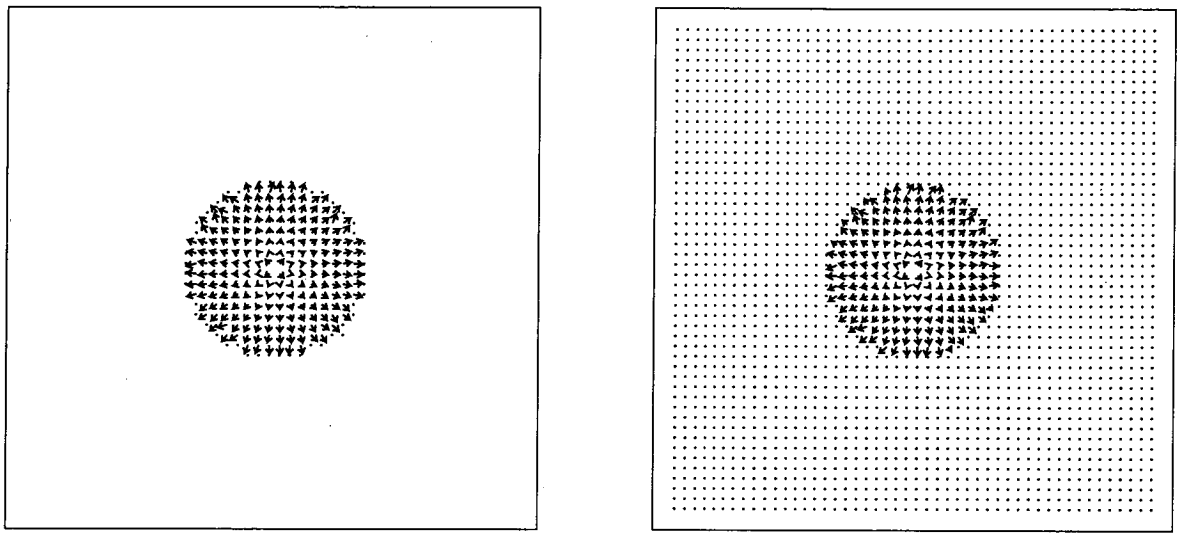
The intensity constraint equation, Eq.(3.11), provides scene information through the gradients of the image intensity function. As the optical flow is constrained in a manner dependent on the motion parameters this constraint can be applied to the image flow. Hence Eq.(3.11) and Eq.(5.17), weighted appropriately by some matrix R, can be solved in a least squares manner,

$$R \begin{bmatrix} I_x & I_y \\ -(D_R M_T y - D_R L_T + B_R M_T) & (D_R M_T x - D_R K_T + A_R M_T) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$
$$= R \begin{bmatrix} -I_t \\ A_R L_T - A_R M_T y - B_R K_T + B_R M_T x \end{bmatrix} \tag{5.18}$$

Provided the a priori knowledge of the camera motion constraint is linearly independent of the intensity constraint equation a solution is possible. Both local and global solutions can be implemented in an identical fashion to that used for the colour algorithms Eq.(5.3) and Eq.(5.5). For the purposes of this work we restrict ourselves to considering a local first order approach only, referred to as method IF10.

5.7 Motion Performance

Image sequences captured or synthesized along with their accompanying motion parameters and optical flow are very difficult to find. This may, in part, be due to the labourious and time-consuming nature of synthesizing such image sequences. An exception is the sphere



(a) (b)

Figure 5.7: *Image flow fields computing using algorithms (a) IF5 and (b) IF10, for the Sphere sequence. Operating parameters set at: differential kernel length of 5 and* $\alpha = 1.5$.

sequence shown in Figure A.7 that illustrates a Lambertian sphere illuminated by a point source located directly behind the imaging sensor. The sensor is initially located at a distance of $1m$ from the sphere and translates along its optical axis towards it at a rate of $5mm$/frame. The focal length is fixed at $8mm$ with a pixel size of $15\mu m \times 15\mu m$.

The image flow field computed using algorithms IF5 and IF10 are show in Figure 5.7. Initial impressions would suggest that little difference exists between the two computed fields. Importantly, however, in the background region the incorporation of the motion constraint has added a degree of stability; the estimates in this region are zero for IF10 whereas it is undefined in the case of IF5. The first two rows of Table 5.1 show the error analysis of the two computed image flow fields against the expected optical flow. It is observed that the incorporation of the motion parameters has provided some enhancement. This is particularly evident in the mean value which shows that IF5 produced a large number of highly erroneous estimates.

The third table entry shows the mean values computed from a sensitivity test where Gaussian perturbations were added to the expected motion parameters. A total of 100 trials were conducted with the standard deviation of the perturbations set at $5cm$/frame for the translation components and 0.05 radians/frame for the quaternion components.

Algorithm	Mean Square Error	Median Square Error
IF5	61.3	8.38×10^{-4}
IF10	2.11×10^{-2}	6.30×10^{-4}
Sensitivity IF10	2.15×10^{-2}	6.30×10^{-4}

Table 5.1: *Square error of computed image flow against expected optical flow for the Sphere sequence.*

These are reasonably realistic values for a robotic arm. A maximum median square error value of 6.32×10^{-4} with a standard deviation of 8.34×10^{-7} was observed. These results would indicate that the solution is largely insensitive to error in the motion parameters. This is not unexpected, as the solution is dominated by the gradient information for this well formed example. Although more extensive testing is required these results suggest that the use of motion parameters may offer significant advantages.

Chapter 6

Analysis on Realistic Image Sequences

The emphasis throughout this thesis has been on the performance analysis and comparison of various image flow algorithms. Most of this work has been presented for relatively simple synthetic image sequences. In this chapter more realistic image sequences are considered with the aim of answering one major outstanding question: how do global gradient methods compare to local gradient algorithms?

Up to this point, from the analysis of less realistic imagery, a conclusion can be drawn that local differential methods, and in particular the first order least squares method (IF5), perform much better than the global method of Horn and Schunck (IF4). However, the selection of alternative operating parameters for IF4, especially the number of iterations, could vary this conclusion. Generally speaking increasing the number of iterations, thereby allowing convergence to the global minimum of Eq.(3.13a), results in an enhanced performance for IF4. As previously noted (Section 3.6) this is made at the expense of the resolution for the computed image flow and the speed at which the field is computed.

Similar conclusions can be drawn regarding the comparative performance of correlation versus gradient algorithms. Local differential algorithms provided superior performance. However, the proponents of correlation algorithms would also strongly argue that their results would be significantly improved by the use of some iterative smoothness constraint.

It is clear, therefore, that the principal issue is whether, or not, a smoothness constraint should be used.

This chapter studies this problem with the aid of realistic image sequences using Horn and Schunck's global gradient IF4 method and the local first order least squares IF5 algorithm. Quantitative analyses of the type presented in Chapter 4 are used. Unfortunately, due to a lack of real image sequences, with accurate optical flows known a priori, many of the results presented are of a qualitative nature.

6.1 Quantitative Analysis

Two image sequences are used for the quantitative analysis: the Yosemite fly-through, see Figure A.12; and Otte and Nagel's block world, see Figure A.13. The Yosemite fly-through sequence is a synthetically generated animation for which the exact optical flow is unknown. Nevertheless, Barron et al.[54] have provided a very good estimate for the majority of the optical flow field, with the exception of the cloud motion at the top the frame. Here the clouds evolve in a complex manner and it is difficult to predict their true optical flow. Table 6.1 shows the results computed using algorithms IF4 and IF5 for the first order differential kernel sizes of 3, 5 and 7. All derivatives were computed using a regularizing spatiotemporal smoothing coefficient of $\alpha = 1.5$. Operational parameters used for the IF4 estimator were $\gamma = 10$ over 5000 iterations to provide substantial smoothness on the estimates. Table 6.2 shows the results computed for Otte and Nagel's block world sequence, a real image sequence captured using a CCD camera mounted on a translating platform. Identical operating parameters for the Fly-through sequence were used for the first order differential kernel sizes of 3, 5, 7 and 9.

It should be noted that many of the same observations made in Chapter 3 and Chapter 4 can be made here. Firstly, the median square error values tend to provide a better estimate of the performance of the algorithms because mean value estimates are dominated by extreme data-points. Examination of the computed image flow fields for each example shown in Figures 6.1 — 6.4 illustrate there are large isolated errors. This is particularly evident in the estimates computed using method IF5. In both image sequence examples the global IF4 method provided a superior performance to the IF5 technique.

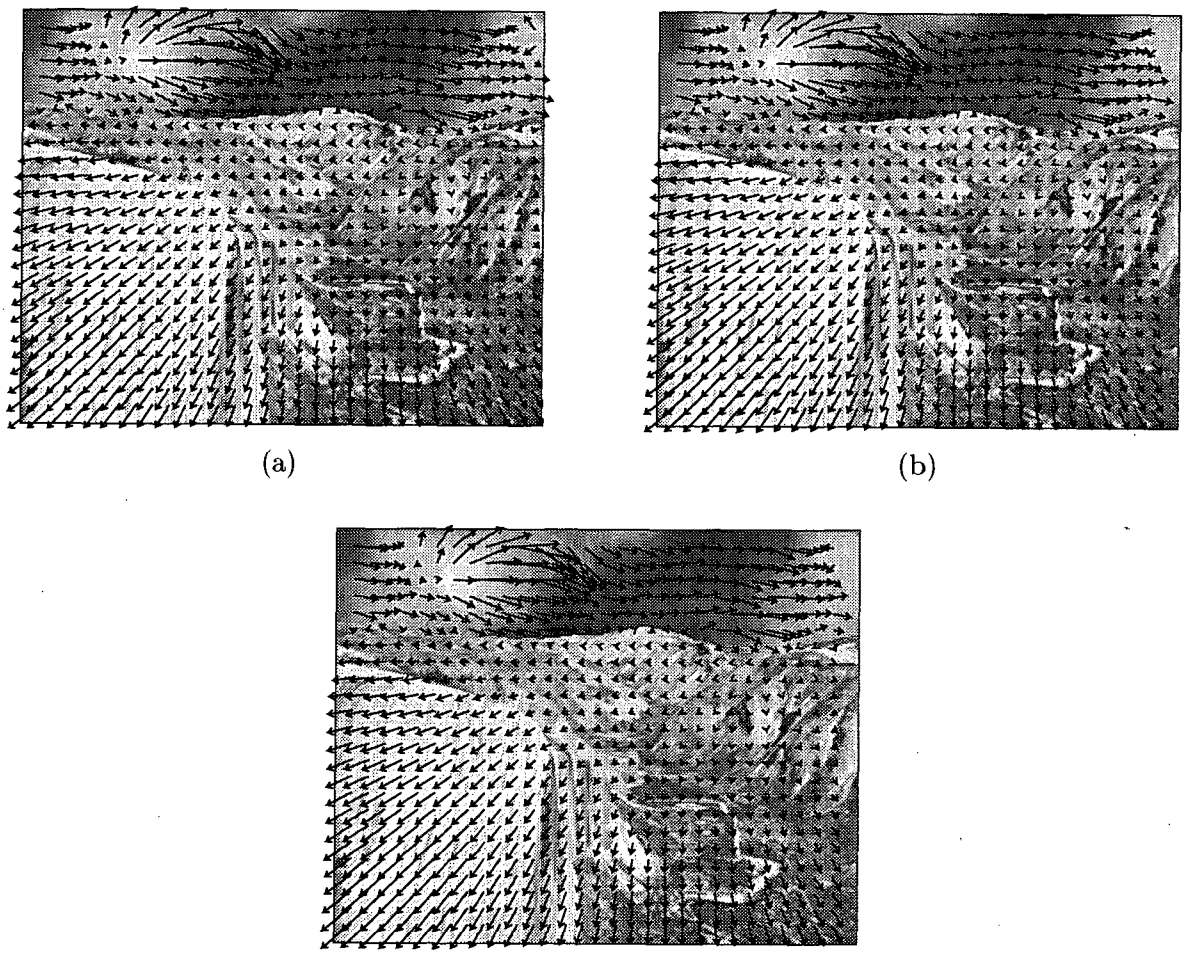
Secondly, it is observed from Table 6.1 and Table 6.2 that the mean square error value

Method	D Kernel Size	Mean	Median	Std. Dev.
IF4	3	0.7226	0.0358	2.17
		7.39°	0.01°	12.9°
	5	0.7020	0.0193	2.18
		6.92°	0.15°	12.9°
	7	0.6914	0.0167	2.18
		6.81°	0.001°	12.8°
IF5	3	1.8580	0.0568	7.03
		11.30°	0.23°	18.8°
	5	1.7227	0.0237	7.66
		9.60°	0.01°	18.2°
	7	1.7336	0.0203	8.03
		9.43°	0.17°	17.6°

Table 6.1: *Yosemite fly-through image sequence. Comparative error metric results for various derivative kernel length (aperture sizes) using the Yosemite Fly-through sequence. Flow fields shown in Figure 6.1 and Figure 6.2.*

Method	D Kernel Size	Mean	Median	Std. Dev.
IF4	3	2.6806×10^{5}	8.2658×10^{-3}	6.81×10^{5}
		10.78°	2.82×10^{-3}°	20.0°
	5	2.6678×10^{5}	4.8328×10^{-3}	6.80×10^{5}
		10.45°	9.10×10^{-3}°	20.2°
	7	2.6549×10^{5}	4.5195×10^{-3}	6.78×10^{5}
		10.40°	2.18×10^{-3}°	20.2°
	9	2.6419×10^{5}	4.4603×10^{-3}	6.77×10^{5}
		10.37°	1.81×10^{-3}°	20.2°
IF5	3	2.6564×10^{5}	2.5103×10^{-2}	6.79×10^{5}
		15.44°	1.23×10^{-2}°	25.8°
	5	2.6300×10^{5}	1.1975×10^{-2}	6.76×10^{5}
		14.14°	3.33×10^{-3}°	25.6°
	7	2.5901×10^{5}	1.0513×10^{-2}	6.71×10^{5}
		13.69°	4.30×10^{-3}°	24.9°
	9	2.5631×10^{5}	1.0498×10^{-2}	6.68×10^{5}
		13.58°	6.15×10^{-3}°	24.3°

Table 6.2: *Otte's Block world image sequence. Comparative error metric results for various derivative kernel length (aperture sizes) using Otte and Nagel's block world sequence. Flow fields shown in Figure 6.3 and Figure 6.4.*



(a) (b)

(c)

Figure 6.1: *Computed image flow using differential IF4 algorithm for the Yosemite fly-through sequence. Operational parameters set at:* $\sigma = 1.5, \gamma = 10$, *5000 iterations, and first order differential kernel lengths of (a) 3, (b) 5, and (c) 7.*

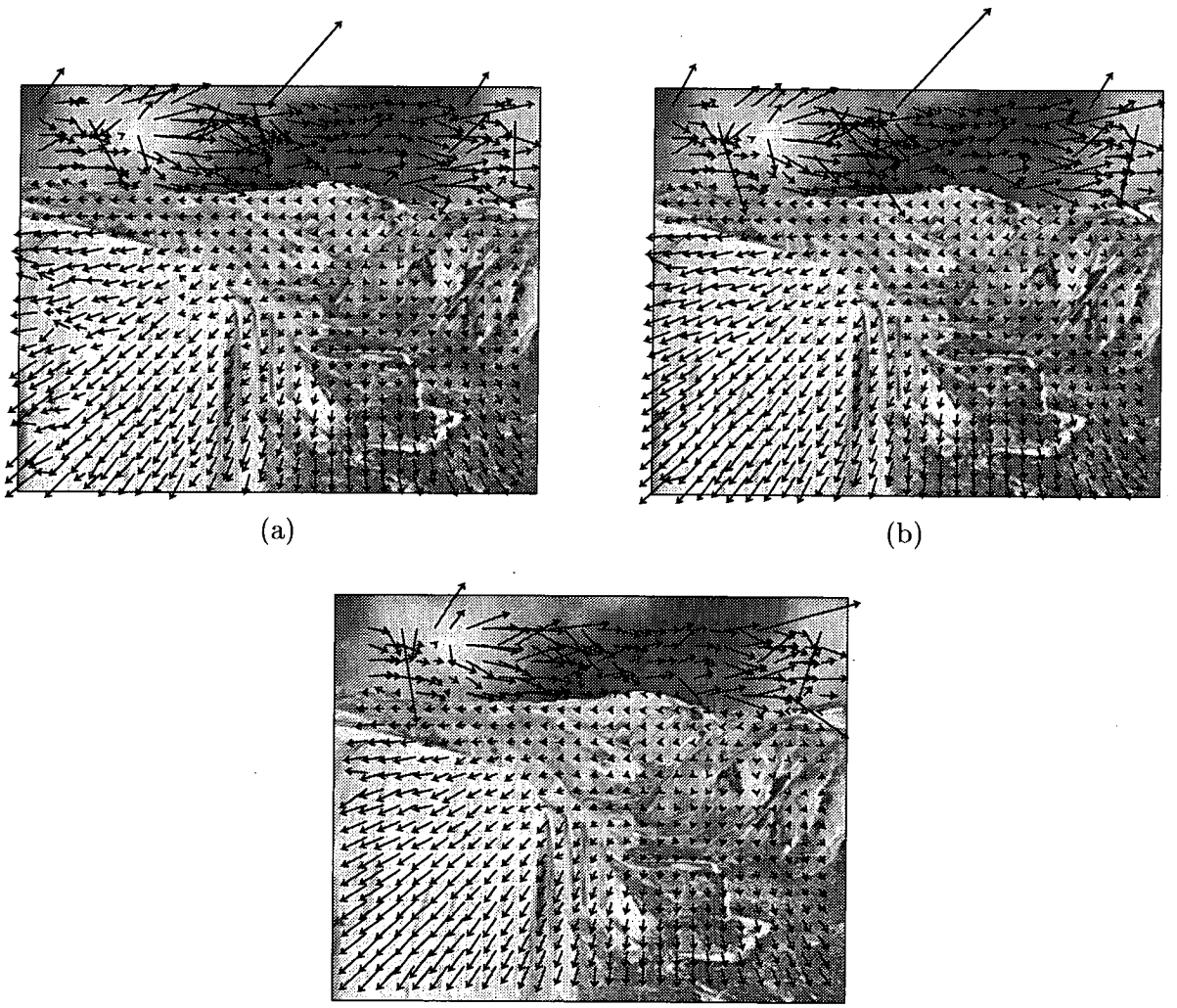
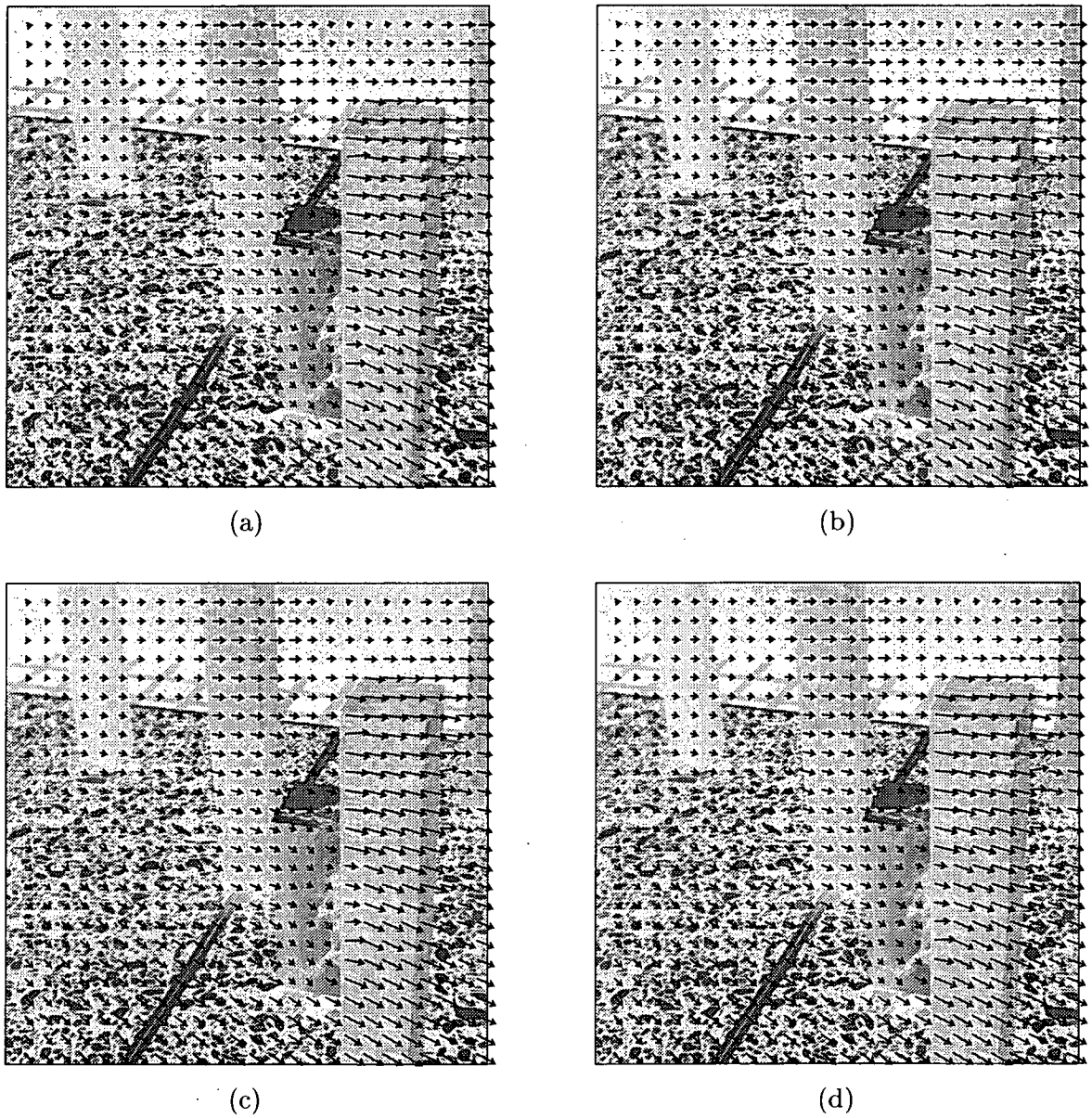


Figure 6.2: *Computed image flow using differential IF5 algorithm for the Yosemite fly-through sequence. Operational parameters set at:* $\sigma = 1.5$ *and first order differential kernel lengths of (a) 3, (b) 5, and (c) 7.*



(a) (b)

(c) (d)

Figure 6.3: *Computed image flow using differential IF4 algorithm for the Otte and Nagel's block world sequence. Operational parameters set at:* $\sigma = 1.5, \gamma = 10$, *5000 iterations, and first order differential kernel lengths of (a) 3, (b) 5, (c) 7 and (d) 9.*

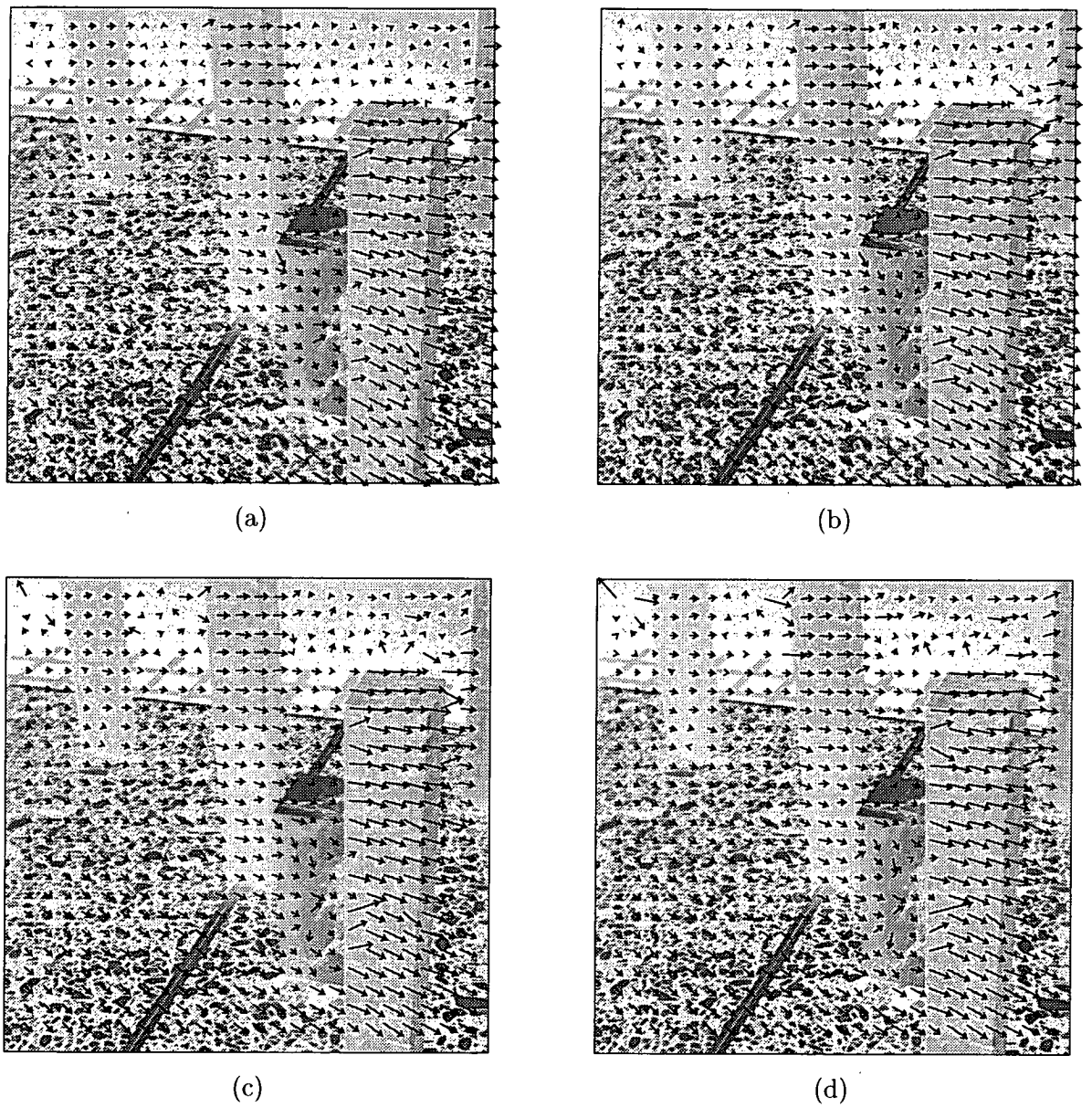


Figure 6.4: *Computed image flow using differential IF5 algorithm for the Otte and Nagel's block world sequence. Operational parameters set at: $\sigma = 1.5$ and first order differential kernel lengths of (a) 3, (b) 5, (c) 7 and (d) 9.*

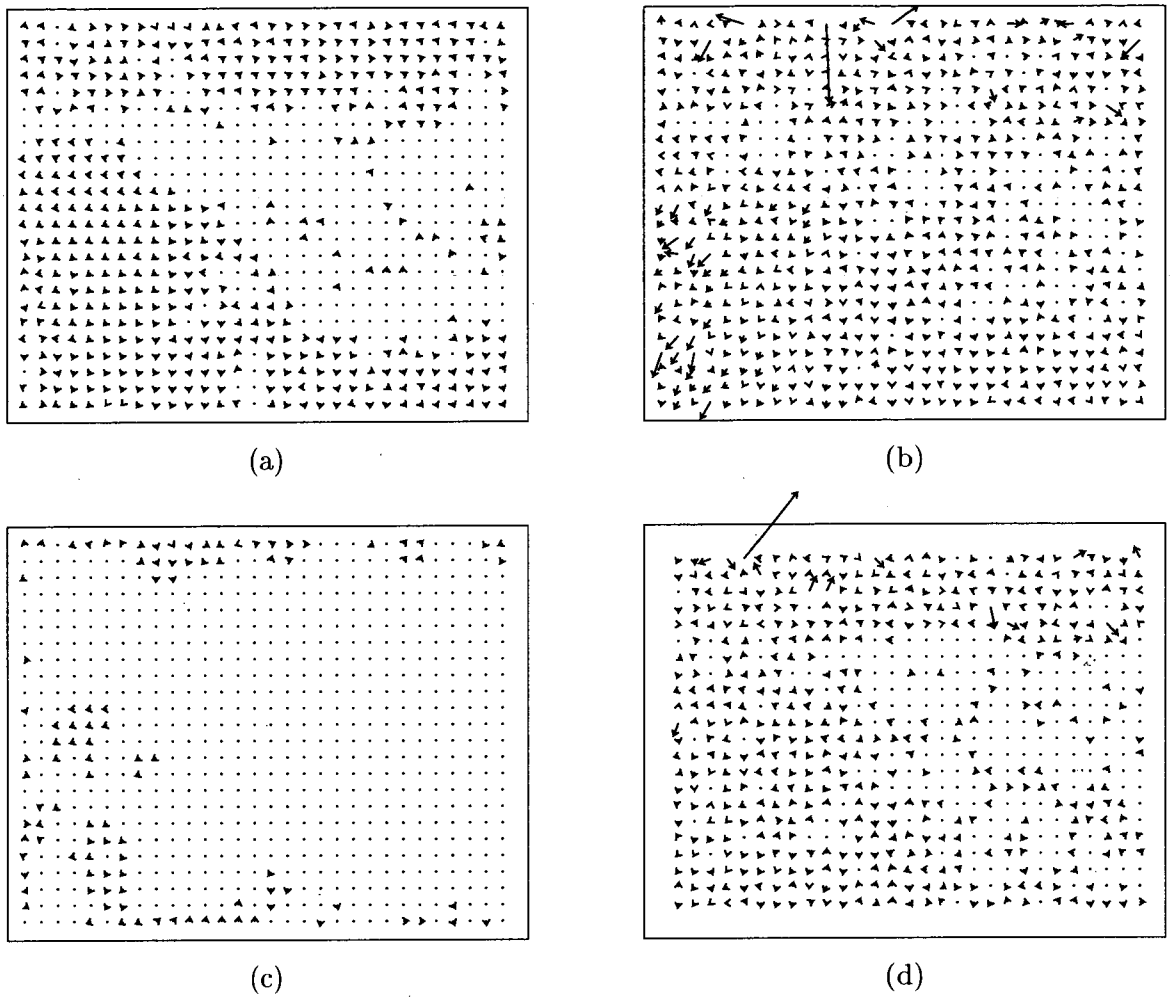


Figure 6.5: *Differences between computed image flow for the Yosemite fly-through sequence shown in Figure 6.1 and Figure 6.2. (a) Difference between IF4 kernels 3 and 5; (b) IF5 kernels 3 and 5; (c) IF4 kernels 5 and 7; (d) IF5 kernels 5 and 7.*

does not change significantly with increasing size for the difference kernel. By contrast the median square error value does improve, especially from a length of 3 to 5 points. This suggests that increasing the length of the kernel does not prevent extreme image flow values from being computed. It is observed that such erroneous image flow points are computed at points in the image where the image intensity gradients are either small, or strongly violate the assumption that changes in image intensity are due solely to camera motion. In these cases little is gained by increasing the gradient kernel size, as the underlying assumption is invalid. This is clearly illustrated in Figure 6.5 and Figure 6.6 where only small differences between computed image flow fields are observed for the increasing kernel length.

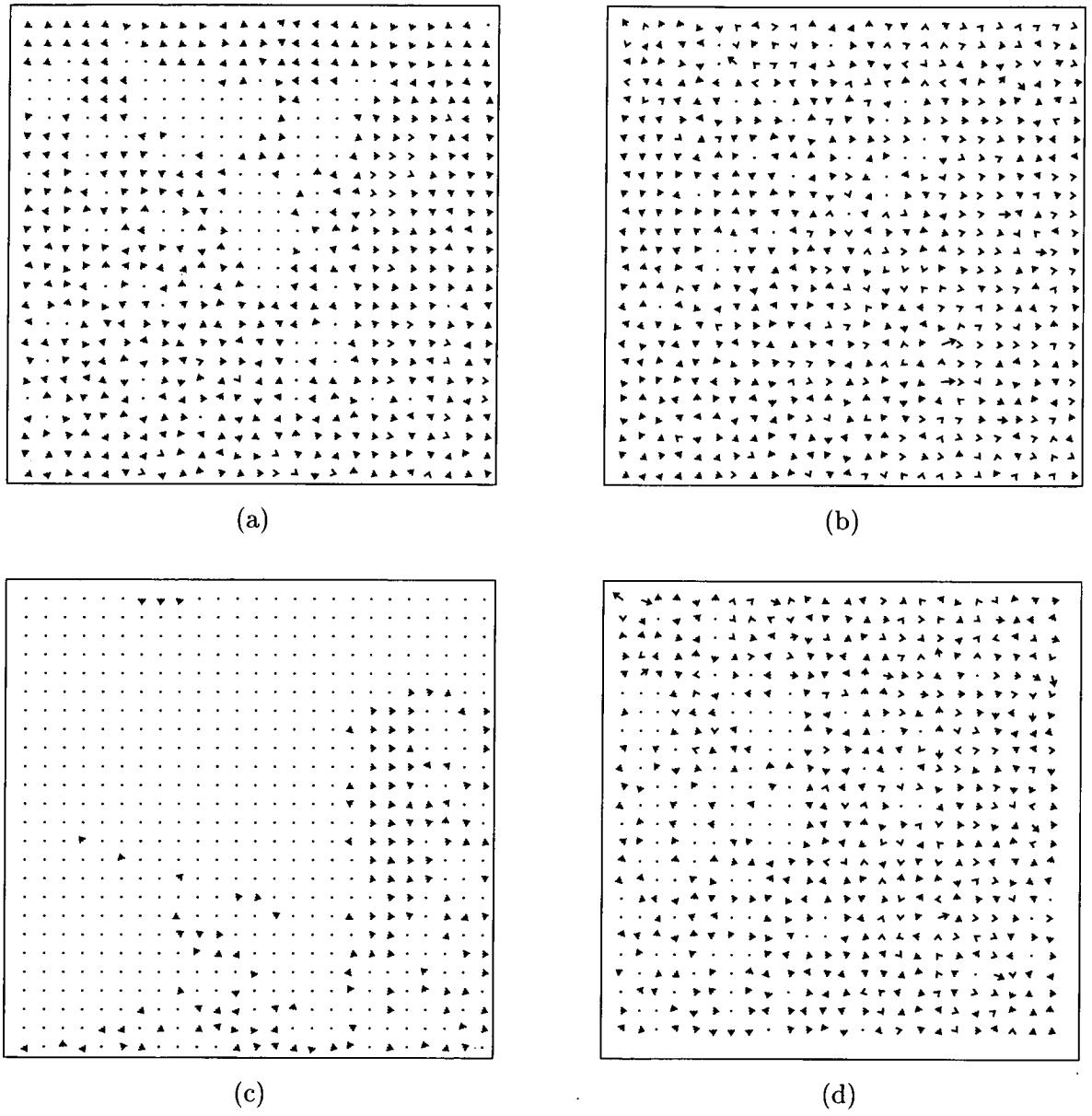


Figure 6.6: *Differences between computed image flow for the Yosemite fly-through sequence shown in Figure 6.3 and Figure 6.4. (a) Difference between IF4 kernels 3 and 5; (b) IF5 kernels 3 and 5; (c) IF4 kernels 5 and 7; (d) IF5 kernels 5 and 7.*

Thirdly, it should be noted that the results provided here are for 100% density image flow fields, no thresholding has been applied. Barron et al.[54] apply a thresholding on the velocity field and reported densities of only 35.1% and 8.7% for their implementations of methods IF4 and IF5 respectively. This thresholding was based on the magnitude of the spatial gradient for method IF4, and inverse minimum eigenvalue for method IF5, as indicators of the performance of velocity estimates. It is not surprising, therefore, that the results presented here would appear less favourable than those reported in Barron et al.. However, it should be noted that our implementation of Horn and Schunck's algorithm performed significantly better than [54].

However, the use of indicators, such as the gradient magnitude or inverse eigenvalue, for thresholding the image flow field is ill advised due to the generally poor correlation between indicator and velocity errors. Table 6.4 and Table 6.5 present the correlation coefficients computed for the various indicators given in Chapter 4 against the square velocity error. The results confirm, with those in Section 4.8, that methods based on the inverse determinant or inverse minimum eigenvalue provide some degree of success. Methods based on the residual or gradient magnitude, however, perform quite poorly especially when used in conjunction with technique IF4. It is also observed that the correlation decreases with increasing differential kernel length and the resultant improvement in the computed image flow. These results are consistent with those observed for the less realistic image sequences. Otte and Nagel[167] present dense flow field results for their block world image sequence computed using methods based on much larger aperture sizes. Our results computed for IF5 are comparable to those they reported.

6.2 Qualitative Analysis

A number of qualitative comparisons can also be made concerning the computed image flows for the two image sequences presented. We note that the two algorithms produce very similar flow fields, except at the top of the image for each sequence. In the case of the Yosemite fly-through sequence the main difficulty is caused by the evolving cloud mass. The global method IF4 is able to reduce the error by the use of the image flow smoothness constraint. This effectively prevents wildly varying velocity vectors from being computed. In the case of Otte and Nagel's block world sequence, IF5 has difficulty in computing a consistent image flow in the bland background regions, particularly the right-hand corner. This problem arises because of a lack of gradient information. IF4 by contrast,

Indicator	Correlation Coefficient		
	3	5	7
Residual	0.41	0.42	0.43
Gradient Magnitude	0.34	0.32	0.32

Table 6.3: *Performance measures of flow field indicators computed using the Yosemite fly-through sequence for algorithm IF4. Implemented with first order difference kernels of length 3, 5 and 7.*

Indicator	Correlation Coefficient		
	3	5	7
Inverse determinant	0.69	0.59	0.49
Inverse minimum eigenvalue	0.70	0.62	0.52
Inverse sum of eigenvalues	0.58	0.46	0.35
Residual	0.26	0.30	0.31
Jähne's measure	0.46	0.44	0.44
Condition number	0.38	0.36	0.34
Sensitivity measure	0.75	0.68	0.60
Gradient Magnitude	0.47	0.39	0.30

Table 6.4: *Performance measures of flow field indicators computed using the Yosemite fly-through sequence for algorithm IF5. Implemented with first order difference kernels of length 3, 5 and 7.*

Indicator	Correlation Coefficient			
	3	5	7	9
Residual	0.02	0.04	0.05	0.05
Gradient Magnitude	0.49	0.49	0.48	0.48

Table 6.5: *Performance measures of flow field indicators computed using Otte and Nagel's block world sequence for algorithm IF4. Implemented with first order difference kernels of length 3, 5, 7 and 9.*

Indicator	Correlation Coefficient			
	3	5	7	9
Inverse determinant	0.72	0.73	0.73	0.73
Inverse minimum eigenvalue	0.50	0.50	0.47	0.44
Inverse sum of eigenvalues	0.70	0.71	0.70	0.69
Residual	-0.12	0.47	0.44	0.55
Jähne's measure	0.33	0.38	0.38	0.35
Condition number	-0.24	0.72	0.18	0.27
Sensitivity measure	0.63	0.69	0.71	0.72
Gradient Magnitude	0.55	0.56	0.54	0.52

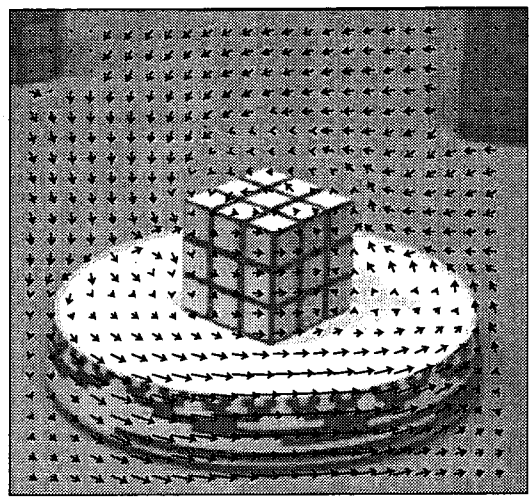
Table 6.6: *Performance measures of flow field indicators computed using Otte and Nagel's block world sequence for algorithm IF5. Implemented with first order difference kernels of length 3, 5, 7 and 9.*

provides an estimate which is smooth but non-zero in this region. Although this at first glance appears a "nicer" solution, it is, nonetheless, no more accurate as the background is actually stationary and the optical flow in this case is zero. Indeed, provided accurate and dense gradient information is available methods IF4 and IF5 produce similar results.

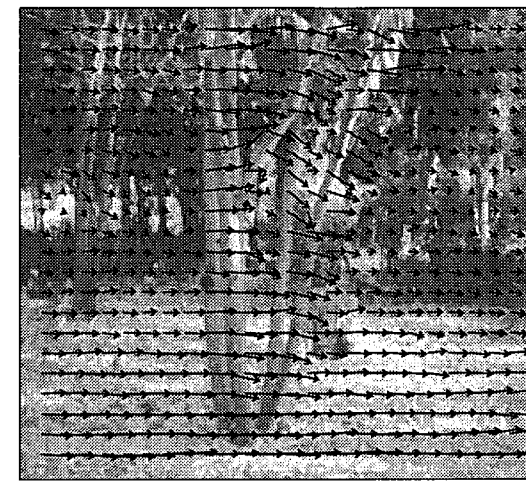
To further highlight the similarities and differences between the two algorithms, four more real images sequences are considered: the rotating Rubic Cube sequence, the planar translating SRI Tree sequence, the diverging NASA Coke Can sequence, and the complex Hamburg Taxi sequence. The computed image flow for algorithms IF4 and IF5 are shown in Figure 6.7 and Figure 6.8 respectively.

In the rotating Rubic cube sequence a cube is placed on top of a rotating table, and is viewed with a stationary camera against a stationary background. Both sequences provide qualitatively acceptable estimates in the region of the turntable and cube, where the spatial gradient is parallel to the direction of motion. Both algorithms also suggest that the axis of rotation passes through the centre of the cube's top face. IF5, however, has difficulties at the base of the turntable where the gradient is orthogonal to the direction of motion. IF4 in this case produces an acceptable flow estimate, overcoming the aperture problem by the use of the smoothness constraint. The difficulties both algorithms have in the bland region of the turntable at the bottom left is evident. In the stationary bland background IF5 does not provide a uniformly zero estimate for the image flow, but an almost random distribution of velocity vectors. The velocity estimates in this region are attributable to noise and ill-conditioning of the problem, a result of estimates made using small image gradients. Method IF4 provides a much more uniform estimate which, although aesthetically more appealing, is no more consistent with the true optical flow and perhaps less detectable than the more random errors of IF5.

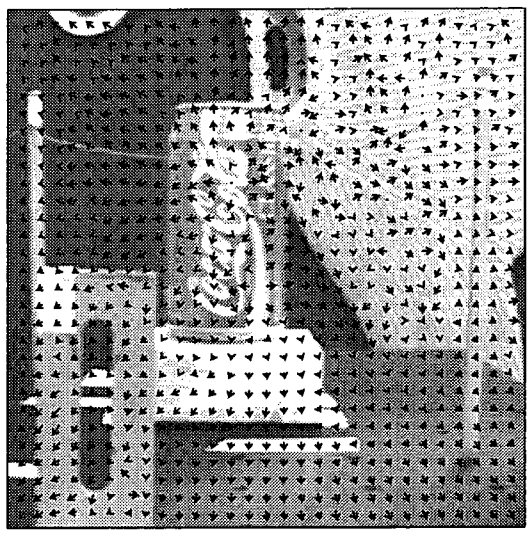
Analyzing the flow fields for the remaining three image sequences reinforces previous findings; that provided there is strong detail in the image both methods will perform well, computing similar results. Barron et al.[54] in their comparative work came to the same conclusion. Where the detail is less and gradient information is poor both algorithms fail, but in different ways. With the global method IF4, smoothing dominates leading to erroneous estimates for the background (as in the Rubic Cube example) or unacceptable interaction between objects in the scene. This is evident in the Hamburg Taxi scene where the initial velocity estimates associated with the car moving to the right and the taxi, mov-



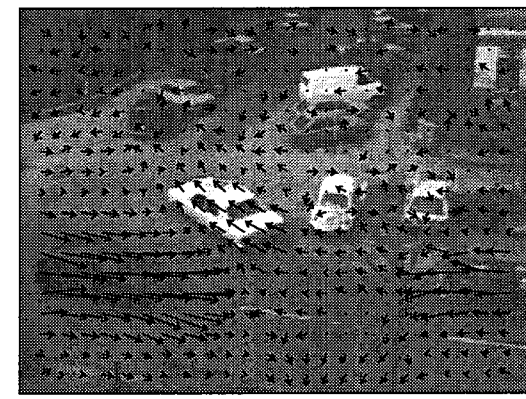
(a)



(b)

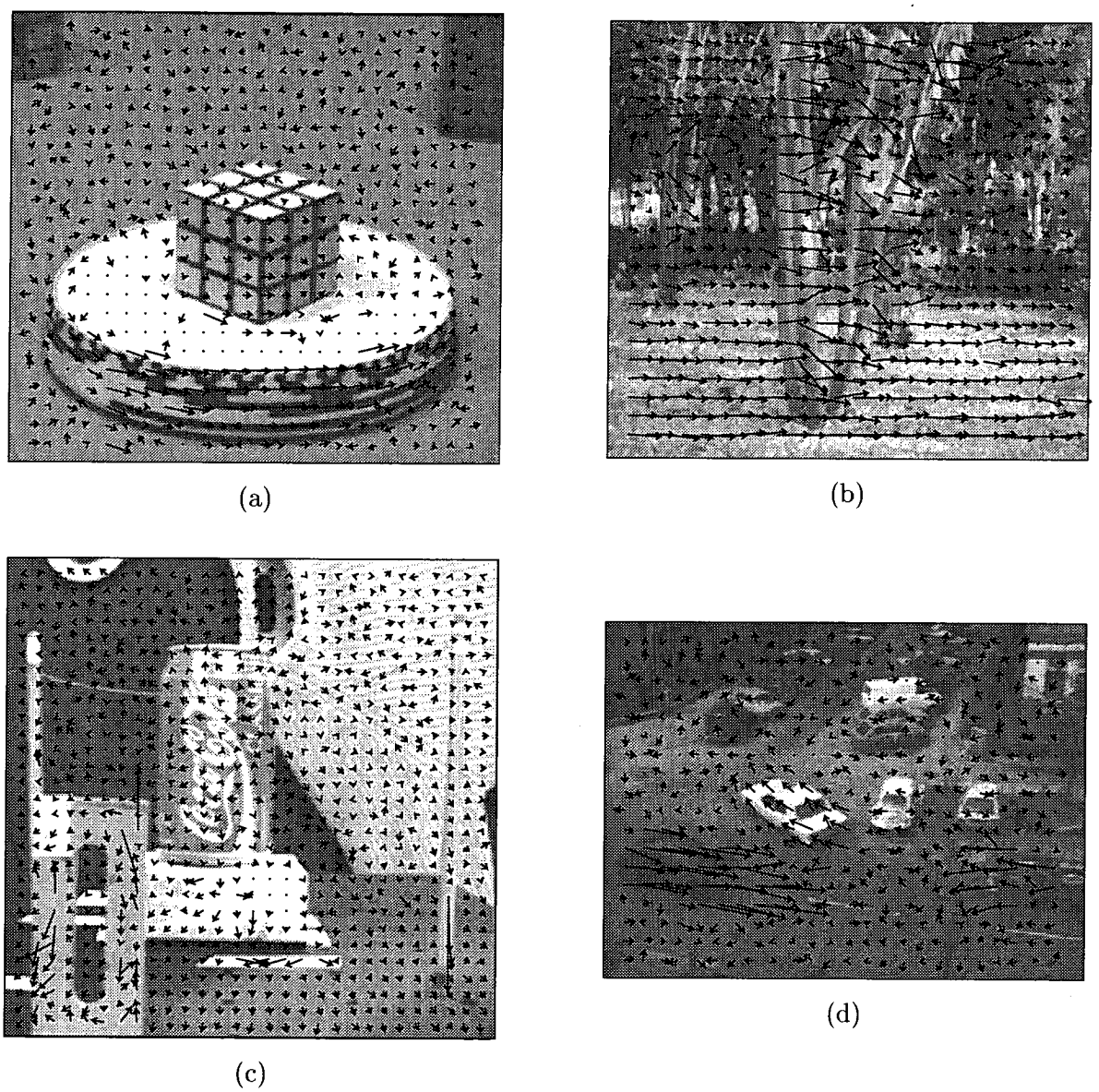


(c)



(d)

Figure 6.7: *Computed image flow using Horn and Schunck's IF4 algorithm, differential kernel length of 5, $\sigma = 1.5, \gamma = 10$ and for 5000 iterations. Sequences: (a) Rubic Cube, (b) SRI Tree, (c) NASA Coke Can and (d) Hamburg Taxi.*



(a) (b)

(c) (d)

Figure 6.8: *Computed image flow using local first order different IF5 algorithm, differential kernel length of 5, $\sigma = 1.5$. Sequences: (a) Rubic Cube, (b) SRI Tree, (c) NASA Coke Can, (d) Hamburg Taxi.*

ing towards the top left-hand corner, interact resulting in a final estimate which suggests the car is expanding. This is not observed in the case of IF5.

On the other hand, the local method IF5 is prone to instability in bland regions. Effective methods for overcoming this are possible; either by a smoothing operation on the computed image flow, some form of smoothness constraint, or by varying the aperture size. A method based on smoothing the flow field after it has been computed is likely to be largely non-iterative, or at least requiring substantially less iterations. Thus it is also less unlikely to present the same problem, such as sensitivity, faced by global algorithms.

Varying the aperture size, however, provides a localized compromise between accuracy and resolution of the image flow. In regions with good gradient information small apertures can be used, while in areas of poor gradient information larger apertures are used. A prime function of an accurate and reliable indicator would be in the selection of appropriate aperture sizes, hence the importance of such a metric. The results given in Table 6.1 to Table 6.6 clearly indicate that a better result using IF5 is feasible, but that reliability metrics are not yet capable of predicting an appropriate aperture size. This is clearly a priority area for future research.

Chapter 7

Conclusions and Future Extensions

In the previous chapters detailed analysis of various image flow algorithms were presented, with particular emphasis on gradient based methods. This chapter summaries the conclusions that can be drawn from this earlier analysis work. These conclusions highlight a number of areas where future enhancements are required. Some preliminary work has already begun, and these areas of additional research are outlined. Finally, the research work of this thesis is reiterated in the context of the broader robotic, machine vision field.

7.1 Conclusions

The analysis work began with a brief review of the various image flow techniques. It was shown that two broad classes can be defined for grouping these algorithms; either as correlation or gradient methods. The algorithms may be implemented either in the Fourier or space-time domains. A number of different error metrics were used in reporting the performances of the various algorithms. These results showed that a measure based on the median square error provided the best insight into the performance of each algorithm. The principal problem with mean error measures occurs when the data set is likely to be dominated by a few very large errors. The median measure is thus a more reasonable descriptor of the data than the mean in these situations.

A brief comparison of iterative methods, closely based on Horn and Schunck's global gradient technique (IF4) and Singh and Allen's correlation method (IF1 — IF3), indicated that no one method was significantly superior. However, comparing the correlation results with the local gradient methods (IF5 — IF7) show that the latter methods compute much superior results at slow speeds. Moreover, if Horn and Schunck's algorithm is not terminated prematurely much better results are obtained, albeit slowly if a small weight is assigned to the smoothness constraint.

Singh and Allen's correlation technique may also be implemented with an iterative smoothness constraint. However, the results computed using the global differential technique clearly indicate a number of shortcomings with such approaches. Firstly, the determination of termination criteria has been largely ignored and ad-hoc approaches used. Such approaches have often resulted in premature termination, causing image flow fields which differ significantly from those if the iterations were allowed to continue towards the global minimum. This, unfortunately, has tended to hide the fact that methods based on an iterative minimization of a smoothness constraint spread velocity information without limit through bland image regions.

Secondly, the resolution of the computed image flow is related to both the number of iterations, as well as the weighting assigned to the smoothness constraint. This interdependence makes it difficult to assess the true resolution of the computed flow field.

Thirdly, global algorithms can be particularly sensitive to errors. Many authors have illustrated this sensitivity, attributing it to poor gradient calculations. Our results also showed that poorly implemented boundary conditions can contribute to the sensitivity of these algorithms. The underlying problem with global algorithms of this type is that the smoothness constraint can spread local velocity errors indiscriminately throughout the entire image flow field.

While the performance of the correlation techniques improved for ever increasing aperture sizes, the same was not observed for gradient techniques. This is attributed to the gradients computed using the larger kernel becoming less representative of the instantaneous derivatives of the data. It could therefore be concluded that correlation methods scale much better than difference methods, i.e. very large window sizes could be used to make very accurate velocity estimates. However, this is made at the cost of resolution as illustrated

by the uncertainty principle (cf. Section 2.5). Correlation approaches are, therefore, often undertaken in a "multi-resolution" way. This issue is discussed in the next section on future research, where a possible technique for multi-resolution gradient algorithms is presented.

A comparison of a number of local differential image flow techniques was also presented. It was demonstrated that all techniques, including first order techniques, rely on the presence of non-zero second order differential terms of the image sensor function in order to overcome the aperture problem. This requirement is explicitly stated in the second order image flow methods, and implicit in both the methods of Sobey and Srinivasan and first order weighted least squares. Moreover, it was shown that the second order and augmented second order methods are limiting cases of the first order weighted least squares technique proposed by Lucas and Kanade.

This comparison of various local gradient methods was based on equal size apertures and illustrated the superiority of the first order weighted least squares algorithm. This enhanced performance over the second order methods can be attributed to incorporating the higher order derivatives of the intensity function in the estimation of motion, while only having to compute first-order gradient terms. The method is also localized and therefore does not present the difficulties associated with a global method, such as Horn and Schunck, of propagating errors throughout the entire velocity field. It is thus suitable for measuring the highest possible resolution in the image flow field.

An investigation into seven performance based indicators for assessing the quality of the image flow estimates was also made. The results showed that these indicators are only generally useful in detecting gross flow field errors. Overall it is argued that methods based on the inverse minimum eigenvalue provide a good combination of simplicity and performance. We have found the first order weighted least squares method is less sensitive to errors in the spatiotemporal gradients than explicit second order techniques. It produces superior results at a price of a poorer correlation between the indicators and error.

Unfortunately, image flow estimates from local differential techniques suffer from a systematic bias. This bias is shown, both by analysis and by simulation, to be in the direction of motion and is attributable to correlations in the finite difference approximations. The bias forms a significant component of the error in the image flow estimate.

The practical issues of regularization of the gradient calculations and the influence of noise corruption were also investigated. It was demonstrated that the SNR of the regularized second order differential terms is not necessarily worse than those of the regularized first order terms. This result is in contrast to others [124,191], who argue that the second order differential terms have a significantly lower SNR to first order terms and consequently second order terms will not result in accurate estimation of the image flow field. Furthermore, both theoretical and practical results indicated that velocity estimation initially improves with increasing levels of smoothing. Excessive smoothing is, however, undesirable not only because it leads to a loss of spatial precision, but also because of the increase in numerical error due to the computation of small gradient terms.

Two innovations involving the inclusion of additional constraints to the under-determined image flow problem were considered. The first investigated colour imagery, the second a priori knowledge of the sensors motion. Both techniques provided information in addition to the intensity gradient constraint.

Three immediate conclusions can be formed from the use of colour imagery. Firstly, chromaticity information increases the robustness of the solution offering a significant improvement by avoiding excessive smoothing, which can often be required for greyscale image sequences. This allows smaller apertures to be considered, resulting in higher flow field resolutions. Secondly, the isoluminance problem, which describes a region of constant intensity but varying chromaticity, can often be avoided by the effective use of chromaticity information. Finally, false motion due to shadows can be ameliorated by reducing the weighting on the intensity information to zero. However, deciding when the intensity gradient is unreliable is a difficult problem that has not yet been solved. Nevertheless, since moving light sources and shifting shadows are a common problem in many machine vision tasks, the results indicate colour will have significant utility.

It has been shown that a constraint, based on the camera motion parameters, can be formed that is independent of scene information. Preliminary results using this constraint, in conjunction with the image intensity constraint, were presented. These results suggest a priori knowledge of camera motion can be used to considerably enhance the stability of the computed image flow.

Finally, a performance analysis of Horn and Schunck's global differential algorithm against

the first order local differential method was made using realistic imagery. The analysis was made in both a quantitative and a qualitative manner. The results clearly demonstrated earlier findings regarding the use of different error metrics and the poor performance of velocity indicators such as the gradient magnitude or inverse minimum eigenvalue. It was also shown that increasing the size of finite difference kernel did little to improve the velocity estimate at points where the estimates are particularly poor.

It was shown that Horn and Schunck's method out-performed the first order local differential method, when compared using 100% density flow fields and the median square error metric. This can be attributed to the superior noise suppression provided by the iterative smoothness constraint on the image flow. Indeed, our qualitative results suggest that in regions of strong detail both methods will perform equally well with similar results. Where the information content is poorer both algorithms fail, with Horn and Schunck's algorithm producing a more aesthetically appealing flow field. However, through the use of smoothing, or variable aperture sizes for the local differential method, it should be possible to implement an algorithm with acceptable performance in areas of weak gradient information. Such an algorithm would also exhibit superior performance to Horn and Schunck's method.

7.2 Future Extensions

It is clear, from the work and conclusions presented, that there is still significant scope for further research work. This includes overcoming limitations posed by the maximum permissible image flow velocity, as presented in Section 2.6 on the sampling theorem of moving pictures. Multi-resolution analysis has been successfully used to overcome this problem for correlation based algorithms, but to date little has been suggested for gradient techniques. In this section some preliminary work into the issue of multi-resolution gradient techniques is presented. This work has a natural extension into the problem of estimating time evolving image flow fields. A common theme to this thesis has been the importance of measuring scene depth and camera motion from image flow. Methods for achieving this are presented here, while broader issues of robotic vision are discussed in the following section.

Multi-Resolution Analysis and Time Evolving Flow Fields

An image sequence captured at a frame rate of ω frames per second with a spatial sampling of (k_x, k_y) has a maximum permissible velocity for the picture elements of,

$$\begin{aligned} |v_N| &= \frac{\omega}{\sqrt{k_x^2 + k_y^2}} \\ &= 1 \quad \text{pixel/frame}, \end{aligned} \tag{7.1}$$

if aliasing is to be avoided (cf. Section 2.6). Picture elements that move faster will be subject to aliasing and consequently incorrectly estimated by image flow algorithms. However, this is a particularly stringent restriction on admissible velocities in many practical applications. Therefore, techniques for effectively increasing the Nyquist velocity are required. This can be achieved in three different ways:

- Increasing the temporal sampling rate
- Decreasing the spatial sampling rate
- Increasing temporal sampling and decreasing spatial sampling

In practice image sequences are often captured at a fixed temporal rate, leaving variation of the spatial sampling rate as the sole practical mechanism for changing the Nyquist velocity. A higher Nyquist velocity requires a reduction in the number of spatial samples, with a consequential loss in spatial resolution of the image and estimated flow field. A less densely packed CCD sensor need not be physically produced in order to achieve a higher Nyquist velocity. Instead a high resolution sensor can be used, with the resulting images being sub-sampled. The image flow is then computed from the sub-sampled images giving a "sub-sampled" image flow. The "full resolution" image flow can then be computed by extrapolation to the higher sampling rate, where "fine adjustments" may also take place, a process known as multi-resolution analysis [186, 198].

In practice a pyramid of images is produced where each higher level in the pyramid is a sub-sampled version of its predecessor, thus a succession of image scales are produced [67, 70, 77, 198]. Image flow is computed from images of the same scale and then projected down to a finer level. Singh and Allen[186] presented a method based on the compact pyramid coding scheme of Burt and Adelson[67]. Singh and Allen also suggest that the

framework of their approach is general and can equally well be implemented as a correlation or gradient technique. Unfortunately, their differential technique can be better classified as a correlation algorithm implemented using spatiotemporal gradient information.

For the local differential methods, image flow is estimated by solving an equation of the form,

$$(A^s)^T W^s A^s (u^s, v^s)^T = (A^s)^T W^s b^s, \tag{7.2}$$

where the superscript s is used to denote the scale at which both the derivatives and image flow are computed (cf. Section 4.1). In a multi-resolution approach the image flow computed at the next scale (u^{s+1}, v^{s+1}) must be computed from the lower scale estimates (u^s, v^s) as well the derivatives A^{s+1} and b^{s+1}. Denoting the residuals of estimating (u^{s+1}, v^{s+1}) and (u^s, v^s) as e^{s+1} and e^s respectively, then the sum of the squared residuals can be expressed as,

$$J = \begin{bmatrix} (e^s)^T & (e^{s+1})^T \end{bmatrix} \begin{bmatrix} (W^s)^{-1} & 0 \\ 0 & (W^{s+1})^{-1} \end{bmatrix} \begin{bmatrix} (e^s) \\ (e^{s+1}) \end{bmatrix} \tag{7.3}$$

where W^s and W^{s+1} are diagonal weighting matrices for the respective data. Minimizing J with respect to the residuals provides a recursive weighted least squares solution for (u^{s+1}, v^{s+1}). As shown by Brogan[34] the recursive weighted least squares solution is,

$$(u^{s+1}, v^{s+1}) = (u^s, v^s) + K^s(b^{s+1} - A^{s+1}(u^s, v^s)) \tag{7.4a}$$

where K^s is a scaling matrix such that,

$$K^s = P^s (A^{s+1})^T (A^{s+1} P^s (A^{s+1})^T + W^{s+1})^{-1} \tag{7.4b}$$

and where,

$$P^s = P^{s-1} - P^{s-1}(A^s)^T (A^s P^{s-1} (A^{s+1})^T + W^s)^{-1} A^s P^{s-1} \tag{7.4c}$$

and,

$$P^0 = ((A^0)^T (W^0)^{-1} A^0)^{-1}. \tag{7.4d}$$

Recursive weighted least squares is also known adaptive Kalman filtering [32]. This technique has also been used for computing a time evolving image flow field [53, 78, 90]. Indeed the same set of equations are required, where the superscript s now denotes time rather than scale. In a practical situation where only a limited number of iterations are possible between

time steps, both time and scale iterations may be intermingled. It should be noted, however, that the velocity estimates from Eq.(7.2) are known to be biased, (Section 4.5). This bias is due to correlation of the errors in computing the spatial and temporal derivatives, and therefore methods that reduce the derivative errors will help reduce the significance of the bias. One possible way of achieving this may be to apply the Kalman filtering to the computation of the gradient terms rather than the image flow velocity estimates.

Depth Estimation

Vision is an indispensable source of information for the operation of mobile robots or autonomous vehicles. The execution of tasks involving search, exploration, or manipulation appear almost impossible without visual support. While the robot is moving, the resulting images acquired by its camera are changing continually, even if the observed environment is completely static. In this case, image motion can be used to obtain useful information about the robot's self-motion and about the three-dimensional layout of the scene.

Stereo vision uses the knowledge about the separation of the two sensors and the disparity between the two views to compute relative depth by triangulation. For a monocular vision system the image flow field provides the measure of disparity, and the camera motion knowledge about the separation of the sensor between the acquired frames within the image sequence [5, 7, 45, 121, 122, 138, 200]. Utilizing the geometric relationship between camera motion and optical flow given in Chapter 5, and in particular the development of the motion parameter constraint equation Eq.(5.17), we recall that,

$$\begin{bmatrix} D_R u - A_R \\ D_R v - B_R \end{bmatrix} Z = \begin{bmatrix} M_T u - K_T + M_T x \\ M_T v - L_T + M_T y \end{bmatrix}. \tag{7.5}$$

This is an over-determined system of equations for the unknown relative depth Z. A solution may be formed by solving for it in a least squares manner,

$$Z = \frac{(D_R u - A_R)(M_T u - K_T - M_T x) + (D_R v - B_R)(M_T v - L_T - M_T y)}{(D_R u - A_R)^2 + (D_R v - B_R)^2}. \tag{7.6}$$

It should be noted, as can be seen in equations 7.6 and 5.15, that scene relative depth can only be estimated provided the camera motion is not entirely rotational.

The least squares relative depth estimate is singular where,

$$\begin{bmatrix} u_s \\ v_s \end{bmatrix} = \begin{bmatrix} \frac{A_R}{D_R} \\ \frac{B_R}{D_R} \end{bmatrix}. \tag{7.7}$$

By substituting Eq.(7.7) into Eq.(5.15) the point in the field of view at which depth cannot be measured is given by,

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} \frac{K_T D_R - M_T A_R}{M_T D_R} \\ \frac{L_T D_R - M_T B_R}{M_T D_R} \end{bmatrix}. \tag{7.8}$$

Close examination of Eq.(7.8) shows that, in the case of translation only, relative depth is measurable at all points except at the focus of expansion. Verri and Poggio and others [58,71,212] illustrate other salient features of the image flow field that can be utilized in the measurement of relative depth and in particular the global motion parameters [71,212,213].

Egomotion and Motion Parameter Estimation

A fundamental assumption made in the preceding discussion, and in Chapter 5, was that the global motion parameters of the image sensor are known. This is a realistic assumption for many robotic tasks. However, in some cases this information is not available, either because the control system is not equipped with an inertial guidance system or, more significantly, the accumulation of position errors from the guidance system renders such measures useless. If we now consider the problem that the motion parameters are not known a priori then we cannot estimate depth. The motion parameters must first be estimated from the image flow using the scene independent motion parameter constraint equation, Eq.(5.17), and unlike relative depth estimation this is a non-trivial problem. In particular it should be noted that the rotational terms A_R, B_R and D_R are quadratics with respect to the four quaternion components and therefore Eq.(5.17) is a cubic expression in seven unknowns and thus very difficult to solve.

Nearly all researchers in motion parameter estimation realize that once one has some information, such as the location of the focus of expansion, or the values of the rotation parameters, all other parameters are easily obtained [7, 44, 106, 112, 123, 197]. A method that assumes the rotational parameters are zero would be quite easy, but is too simplistic for most real applications. Nevertheless, there is considerable motivation for separating the flow field due to rotational parameters from the flow field due to translational parameters. Verri et al. [58, 71, 213] investigate salient features of the image flow field which can be utilized to separate the translational and rotational components of the image flow. The partial derivatives of a function at its stationary points give an indication of the form of the function at that point. For a one-dimensional signal, the partial derivatives indicate whether the stationary point is a maxima, minima or inflection point. For the optical flow

field the partial derivatives are given by,

$$M = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} \end{bmatrix}. \tag{7.9}$$

In this case, for a two-dimensional signal, the eigenvalues of the matrix M indicate the form of the stationary point:

- Both eigenvalues are *real* and equal. The point is a symmetric maxima or minima and is called a focus point. For positive eigenvalues the point is a source (maxima) or focus of expansion. For negative eigenvalues the point is a sink (minima) or focus of contraction. See Figure 7.1(a).
- Both eigenvalues are *real* and of equal sign. The point is said to be an asymmetric maxima or minima or node point. For positive eigenvalues the point is a source node. For negative eigenvalues the point is a sink node. See Figure 7.1(b).
- Both eigenvalues are *real* and of opposite sign. The point is a saddle point. See Figure 7.1(c).
- The eigenvalues are an *imaginary* conjugate pair. The point is a centre about which the flow field rotates. The magnitude of rotation is given by the magnitude of the eigenvalue, but direction is not specified. See Figure 7.1(d).
- The eigenvalues are a *complex* conjugate pair. The point is a spiral, this being the sum of a focus and a centre point. See Figure 7.1(e).
- All other solutions are infeasible.

Other useful methods for separation the components of the image flow include the differential techniques of divergence and circulation (curl) [7, 43, 44, 48, 65, 66, 75, 76, 87, 103, 106, 112, 123, 134, 149, 151, 152, 195, 197, 220–222, 231, 232, 234].

7.3 Related Issues

In order for the robotics field to truly advance, the problem of a lack of sensory ability must be adequately addressed. This thesis, in part, has addressed the issue of robotic vision and in particular the problem of accurately estimating image and optical flow. Measurement

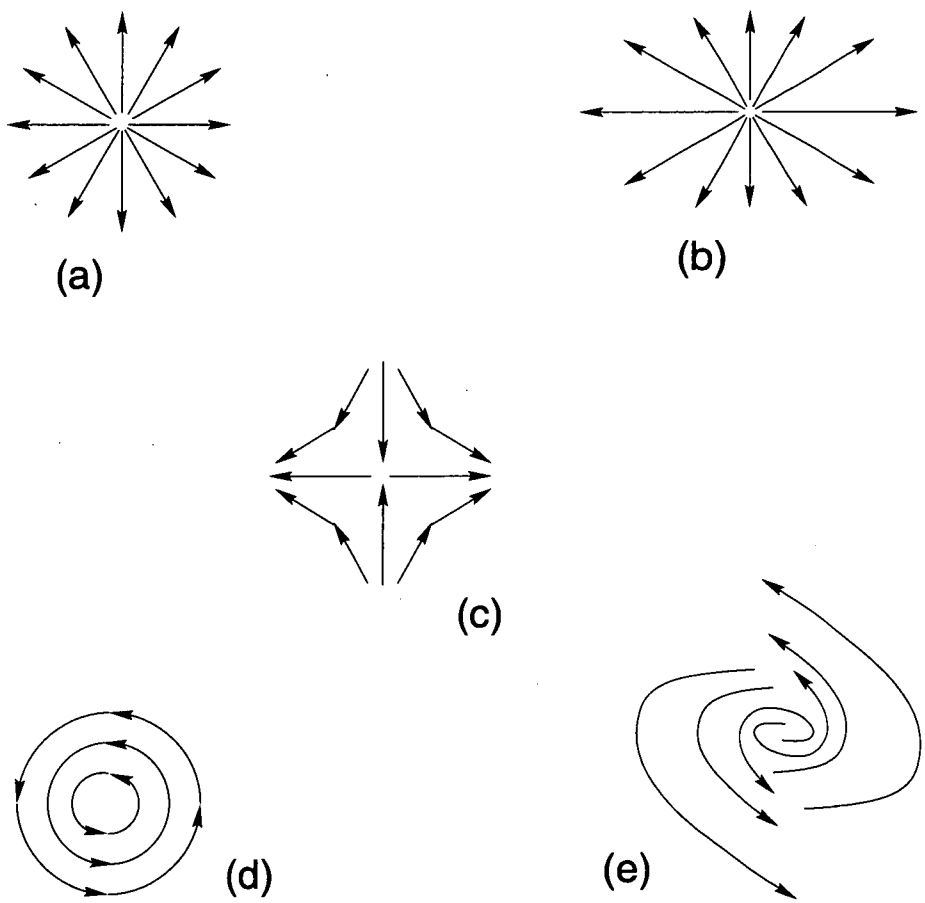


Figure 7.1: *Stationary points in the optical flow field: (a) focus point, (b) node point, (c) saddle point, (d) centre point and (e) spiral point.*

processes are of course fundamental to any control system, as accurate measurements help provide robust control, and hence the motivation for this work. The discussion of actual control systems and their design has, however, not been presented up to this point. This final section presents a brief overview on topics related to control processes based on image measurement. The subject is enormous by virtue of its cross-disciplinary nature, bordering on the fields of psychology, philosophy, artificial intelligence and expert system design. As a result this review should be read as a suggestion of likely starting points, rather than a brief and comprehensive presentation. Topics briefly covered include:

- Vision system design
- Passive vision systems
- Multiple visual cues
- Active vision systems
- Path planning and navigation
- Calibration
- Segmentation of the motion field

The usual recommended approach for solving computer vision problems follows the approach outlined by Marr in his seminal work on computer vision [9]. The problem is iteratively sub-divided into smaller sub-problems or functional components in a top-down fashion, an approach often employed in the solution of many complex information systems [40]. Methods of estimating image flow presented in this thesis represent one such functional component, which are often referred to as a visual cue. Each component represents an autonomous unit and is designed to ensure maximum generality (can be applied to many problems) with minimal physical constraints. An example of a physical constraint would be; "a requirement that only one object of interest be in the field of view". Physical constraints are added at a higher level and are thus integral to a specific application. This approach affords a greater understanding of the vision problem, a greater ability to test the application and assess it limitations, and a greater degree of robustness with the potential for a less restrictive operating requirement [34].

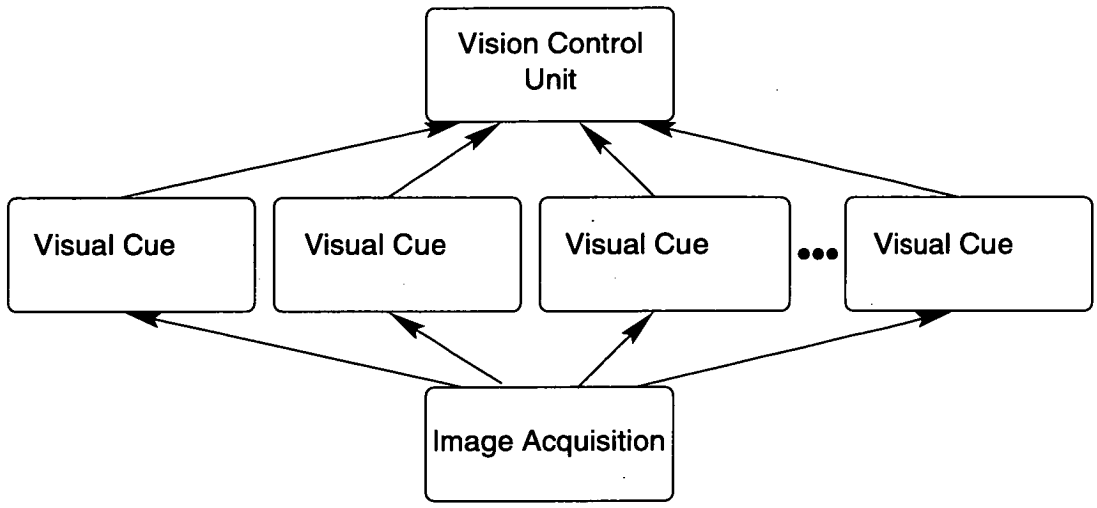


Figure 7.2: *Functional model of a passive vision system.*

The control unit ties together each of the visual cues and makes process dependent decisions based on the measurements provided by these units. Numerous vision systems in current industrial sites are defined as passive, as illustrated in Figure 7.2. In a passive system the imaging sensor is fixed and the flow of information is from sensor through the visual units to the control unit, no feedback is provided. Process decisions are therefore made in an open-loop control configuration, and in order to retain stability the environment must be highly constrained. Examples of current applications for such systems include visual quality control, such as those used in circuit board fault detection and fruit quality assessment, and log orientation for wood milling [3, 5–7, 109, 110]. In each of these cases the camera is fixed in position with an appropriate and fixed lighting system. Objects of interest are then presented in the field of view of the sensor at the correct focal distance. The performance of such systems are tuned by hand and often vary from site to site.

Further improvements can be achieved by increasing the number of visual cues measured, albeit at the cost of additional complexity of the control unit and the overall system. In order to present a single overall "best picture" to the control unit data fusion is employed. The fusing process is quite literally an integration process, where different relative weights are assigned to each of the input sources (visual cues) depending upon their relative reliability and other heuristic measures [46, 80, 85]. This again reinforces the importance that measurement processes provide reliability and confidence estimates. For example, in an autonomous vehicle problem depth estimates may be provided by a number of shape from x algorithms, such as motion, focus, shading and stereo. We can then use our integrated infor-

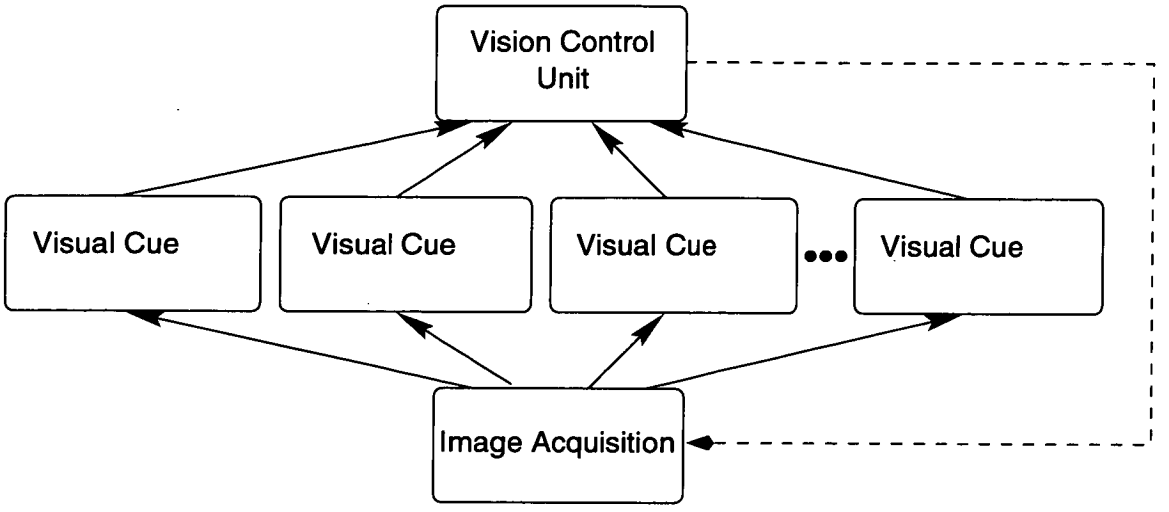


Figure 7.3: *Functional model of an active vision system.*

mation system to make other critical process decisions, such as path planning for collision avoidance or to adaptively investigate a scene.

In an active vision system the image sensor properties are allowed to vary (in a controlled way) and thus provide an additional feedback between the image acquisition and control unit through the image analysis processes, Figure 7.3. Process decisions are therefore made in a closed loop configuration, which provides a significant degree of stability [12, 114, 148, 178]. For a robotic vision system based on stereo images, scene depth is estimated by triangulation of the measured disparity between the image pair and the known imaging geometry. Many such algorithms have been developed, each assuming that the images are acquired from known viewpoints with compatible camera orientations and, of course, with the area of interest in proper focus. However, for real scenes that are deep and wide, no single imaging configuration can obtain stereo images of the entire scene suitable for surface reconstruction. This is because the cameras capture visual fields of limited size and depth. To reconstruct the surface of an entire scene, the imaging configuration must be varied to sequentially capture different parts of the scene. Like human eyes, the cameras must pan and tilt, converge and diverge, and focus on near and far objects [85]. Active vision systems therefore measure changes in time. Hence, by definition, vision systems employing image flow visual cues are active simply because there is a dependent temporal interaction of the camera(s) and scene.

An inevitable requirement of active vision systems is that they merge the data obtained from different imaging configuration, so as to construct a cumulative understanding of the environment. In the case of depth estimation, the surface reconstructed for a given part of the visual field must be added to the cumulative surface data. The reconstruction takes place in a cooperative and interleaved manner; newly acquired data is merged into a cumulative database, which in turn must be used to predict the new imaging configuration and that part of the visual environment to observe next. An immediate question that arises is; "given our current understanding of the scene, what is the optimal reconfiguration of the camera(s) so as to maximize the information obtained in the next look?" The question of planning the next view forms the basis of much active vision research [148,178,223].

In robotics five principal questions about the action to be undertaken must be asked:

- why should the action take place?
- where should the sensor be moved to?
- when should the action take place?
- how should the action be performed?
- what to do when we get there?

The solution can range from the simple; "go to X moving in a straight line then wait for the next instruction, do this action now because I command it," to more complex systems based on fuzzy logic, artificial intelligence and expert design systems, [5,7,39,61,145]. These questions are now not only addressed by the desire to maximize the amount of information and knowledge received in any one look, but also to assist in the problem of navigation and collision avoidance. Navigation requires accurate knowledge of one's location, relative motion and the depth of surrounding objects. Such information can be obtained from the image flow field, inertial navigation devices, and by estimating the sensor's egomotion (self-motion).

Indeed a pragmatic side effect of active visual control is a continuous degradation of calibration due to inaccuracies in the mechanical control system. In order to overcome this problem frequent re-calibration is required, a process that is often quite elaborate and requires special calibration patterns or objects to be viewed [60,64,73,100,104,136,176,184,215]. This is, of

course, not feasible for most practical systems and what is needed is an "on-the-fly" adaptive self-calibration process. The estimation of the global motion parameters from image flow data represents one such group of techniques. Other researchers have proposed integrating system calibration with the surface reconstruction, thus obviating the need for frequent calibration processes [46,144,146,148,233] . Effectively, in such adaptive calibration, the role of the external calibration pattern is fulfilled by either the partially reconstructed surface map of the scene or the image flow field.

The computation of the global motion parameters is a non-trivial exercise. Often the problem is simplified by the assumption that the scene is static. In dynamic environments, however, the appearance of many individually moving objects in the scene adds another level of complexity. Object motion and camera motion now interfere and moving objects may not even cause any image flow at all. The environment cannot be treated as a single rigid object but possibly as several objects. Segmentation of the image flow field according to each object then allows the motion parameters of each segment or sub-field to be computed separately [11,61,63,69,81,82,113,127,139,153,201,205,224]. Moreover, early segmentation of images can lead to enhanced image flow estimates, especially when implemented in an iterative scheme [7].

This thesis has presented a variety of methods for measuring image flow together with a detailed discussion of their characteristics, relative performances and limitations. A number of key issues, improvements and extensions still remain outstanding. These were outlined in the preceding two sections, however in this last section it is clear that one issue in particular requires urgent attention: accurate reliability estimates of the image flow. Such estimates are required at decision levels within a practical vision system. It is required in the data-fusion process that determines the best overall picture of the scene, its structure and motion. It is required in assessing camera motion, navigation, where to look next, how to move, and the calibration of motion. It is also integral to image segmentation and methods of improving image flow computation. Thus, it must be the next step taken towards our long term goal of implementing a robotic vision control process.

Appendix A

Image Sequences

A number of different image sequences were employed for the purposes of testing and comparing various image flow algorithms. Most of these sequences are either "standard" test sequences freely available by anonymous FTP on the *internet* or are easily synthesized. Samples frames from the test sequences are shown in this appendix.

A.1 Synthetic Test Sequences

Numerous synthetic test image sequences were used in this work. These sequences were generated using a variety of methods including: ray-tracing based on the image formation process, simulated camera models and Fourier phase shifting of static images. Fourier phase shifting allows for the production of sub-pixel movement between frames, and is a technique suitable for planar translation motion only. It results in a uniform displacement for each pixel across the entire image plane irrespective of the relative depth of each imaged point. This is physically reasonable provided the field of view is very shallow, i.e. the relative distance between the closest and furtherest points from the sensor focal point is small compared to the average distance of points from the sensor.

In order to produce sub-pixel movements between frames the n^{th} frame of the sequence was generated using,

$$I^n(x,y) = I^0(x-un, y-vn) = \iint \mathcal{I}(k_x,k_y)e^{j(k_x un + k_y vn)}dk_x\,dk_y, \qquad \text{(A.1)}$$

where $\mathcal{I}(k_x, k_y)$ is the Fourier transform of the initial frame $I^0(x, y)$, (k_x, k_y) the spatial frequency components, (u, v) is the velocity in pixels per frame and n is the frame number [21,29].

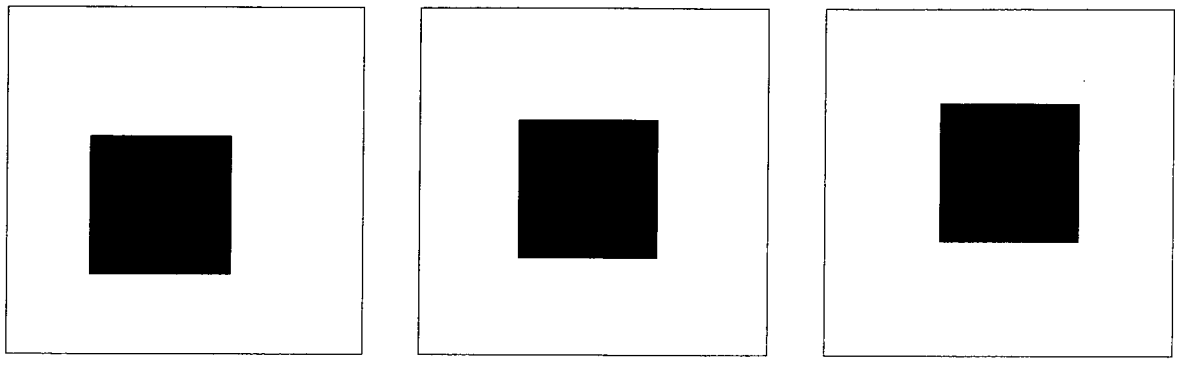
Table A.1, below, lists the synthetic sequences used, including the method by which they were generated, the speed range, who generated the sequence and where they may be obtained from.

Table A.1: Summary of synthetic test images used.

Sequence	Comments and Speed Range
Moving Square Sequence	Large dark square moving on a light background, created by Prof. John Barron, Dept. Computer Sci., University of Western Ontario, London Ontario Canada. Email: barron@csd.uwo.ca, FTP: ftp.csd.uwo.ca Uniform velocity of all points $(1, 1)$ pixel/frame.
Translating Dot Sequence	A single dark dot moving on a uniform light background. Created by the author. Uniform velocity of $(1, -1)$ pixel/frame
Translating Dot on a Patterned Background	A small dark 3×3 square with darker centre translating over a stationary random patterned background. Signal power of square $\times 2$ peak power of background, signal power of square centre $\times 4$ peak power of background. Created by the author. Uniform velocity of $(1, -1)$ pixel/frame
Random Field Set	Fourier phase shift of static random pattern with uniform velocity across the entire image plane field. Created by the author. Any speed possible, though generally in the range 0 ... 2 pixel/frame
Modified Translating Tree Set	Fourier phase shift of frame number 20 from Translating Tree sequence, with uniform velocity across entire field. Created by the author. Any speed possible, though generally in the range 0 ... 2 pixel/frame

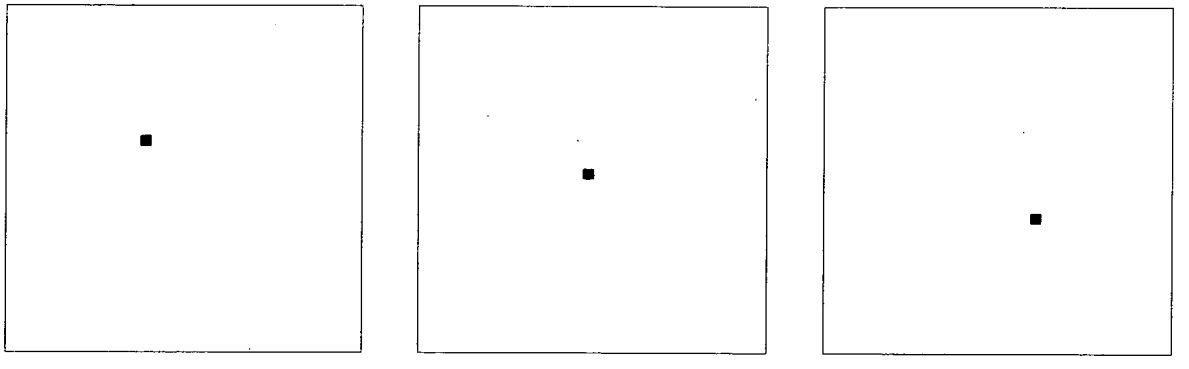
continued on next page

Sequence	Comments and Speed Range
Translating Tree	Created by David Fleet, Dept. Computer Sci., Queen's University, Kingston Ontario, Canada, using simulated camera model. Made available by John Barron. Camera moves normal to line of sight in X-direction. Velocity in x-direction ranging from 1.73 to 2.26 pixel/frame
Diverging Tree	Created by David Fleet using simulated camera model. Made available by John Barron Camera moves along line of sight in Z-direction, focus of expansion at centre of image. Velocity range from 1.29 (left) to 1.86 (right) pixel/frame
Diverging Lambertian Sphere	Created by author using ray-tracing technique. 0 through to 4 pixel/frame
Strips Colour Test Sequence	Created by author. Random intensity images consisting of alternating red and green strips. The strip pattern translates horizontally at 1 pixel/frame, the intensity at 0.3 pixel/frame vertically.
Screwdriver Test Sequence	Created by author. Realistic colour image sequence. 1 pixel/frame vertically.
Poster Test Sequence	Created by author. Complex image with extensive colour detail. However, most gradient information is orthogonal to the direction of motion. 2 pixel/frame vertically.
Shadows Test Sequence	Created by author. Synthetic image sequence of vertically ramped intensity function translating vertically, horizontally ramped saturation function and an alternate hue function. A shadow bar can be seen translating horizontally across the image (affects intensity function only). 1 pixel/frame vertically, and 2 pixel/frame horizontally



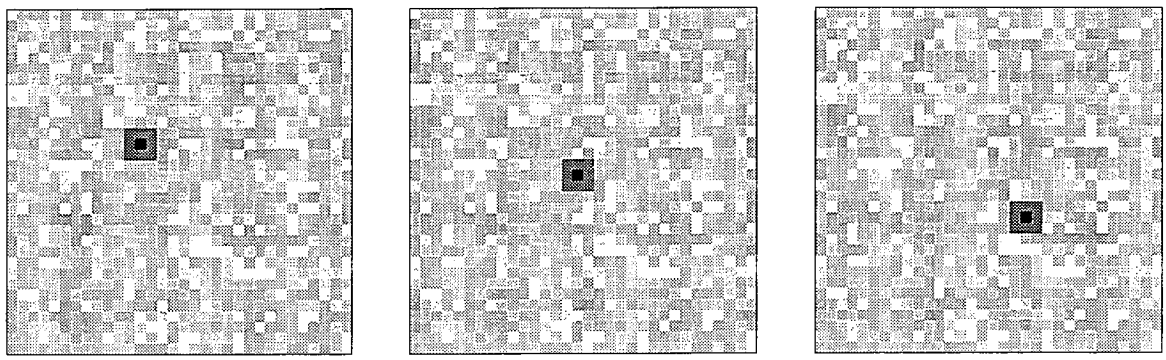
(a) (b) (c)

Figure A.1: *Frames 5, 10 and 15 from the "Square" sequence. Image size is* 100×100.



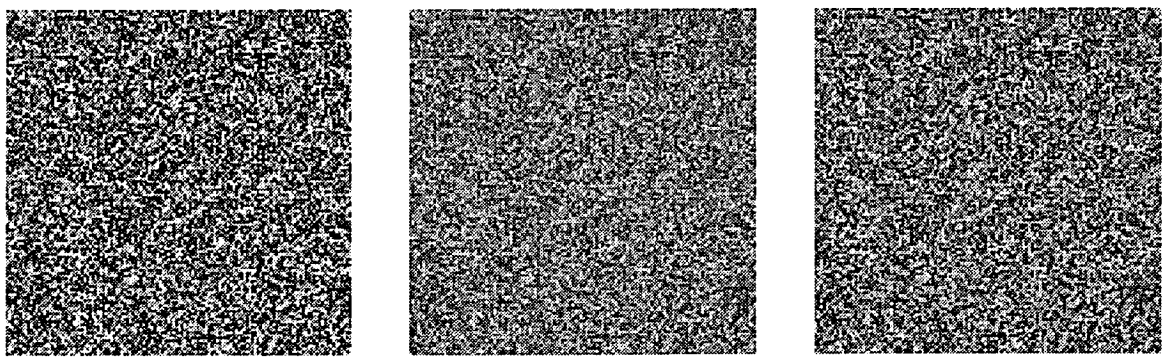
(a) (b) (c)

Figure A.2: *Frames 4, 7 and 11 from the translating "Dot" sequence. Image size is* 32×32.



(a) (b) (c)

Figure A.3: *Frames 4, 7, 11 from the translating dot sequence on a patterned background — "Random Dot" sequence. Image size is* 32×32.



(a) (b) (c)

Figure A.4: *Frames 0, 16 and 32 from the horizontal 0.3 pixel/frame "Random Field" sequence. Image size is* 256×256.

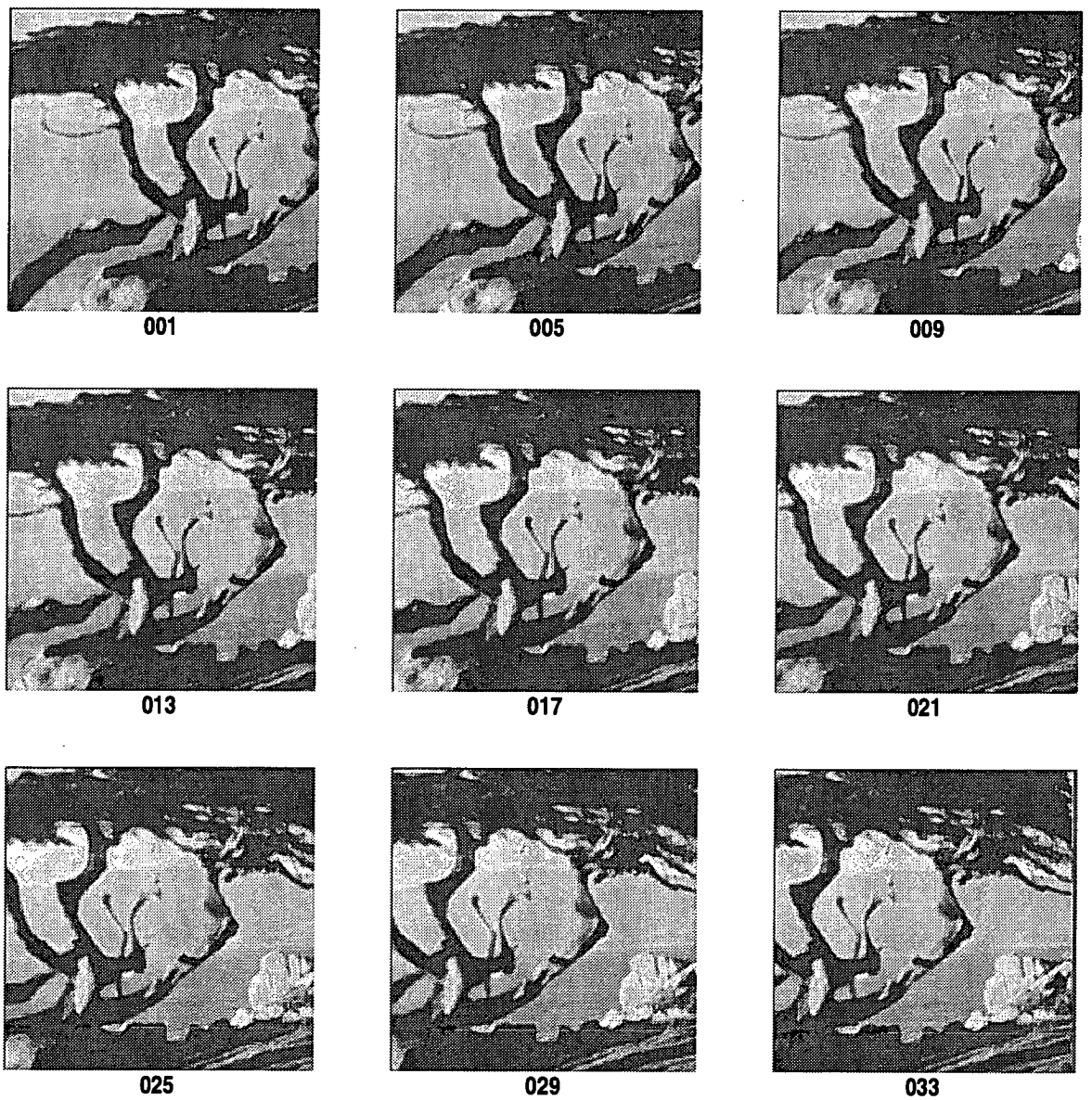


Figure A.5: *The "Translating Tree" sequence. Image size is* 150×150.

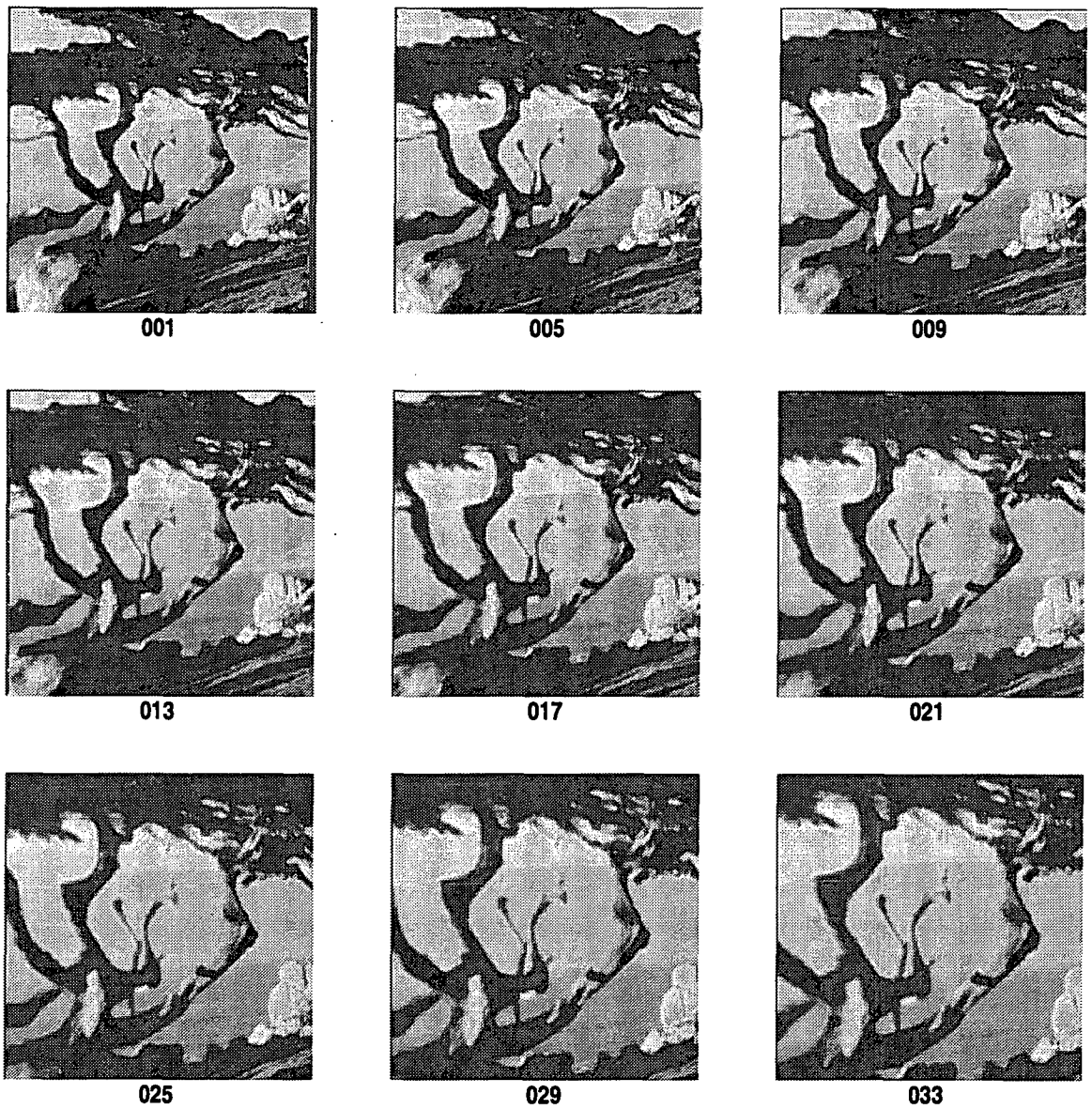


Figure A.6: *The "Diverging Tree" sequence. Image size is* 150×150.

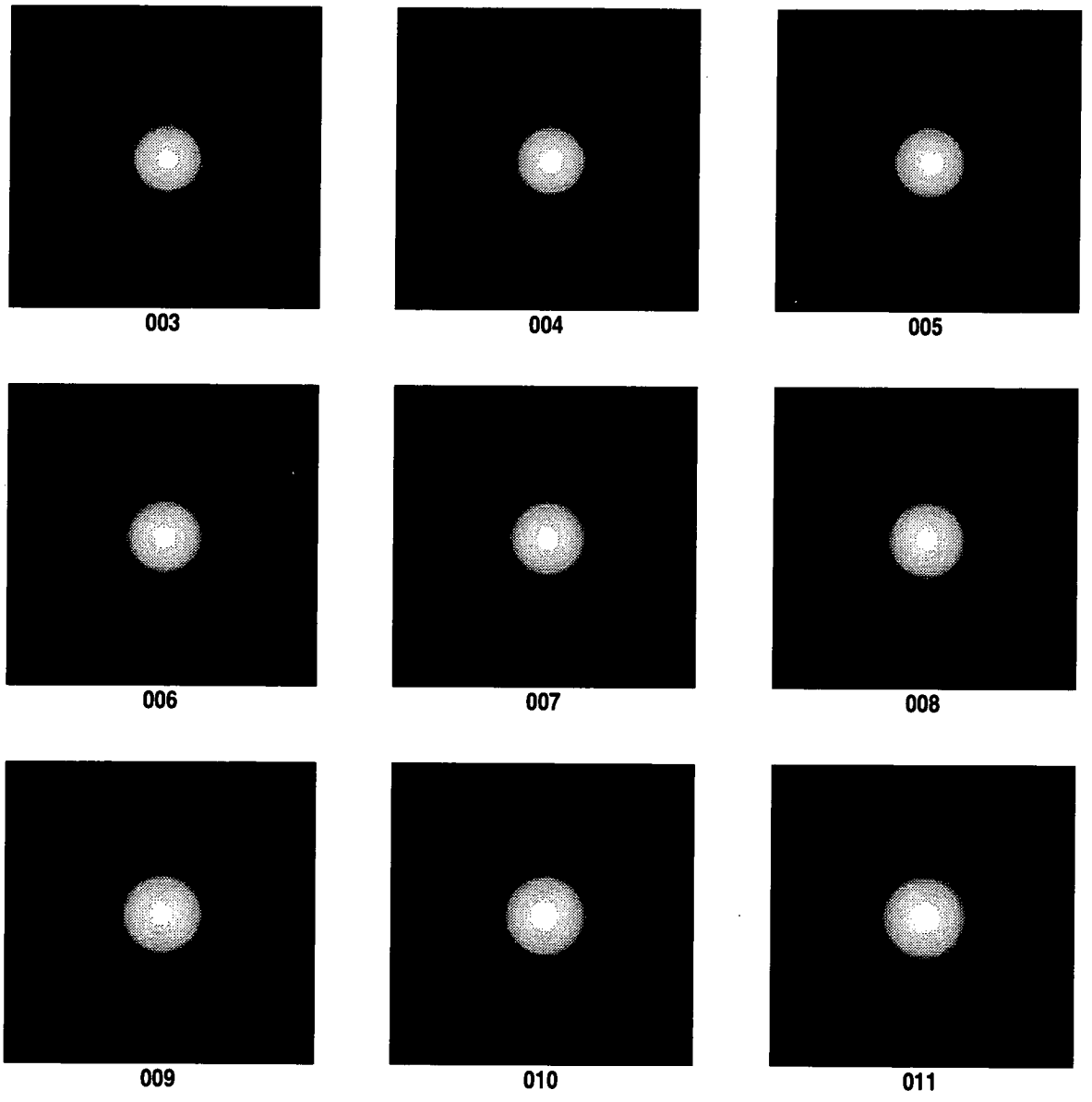
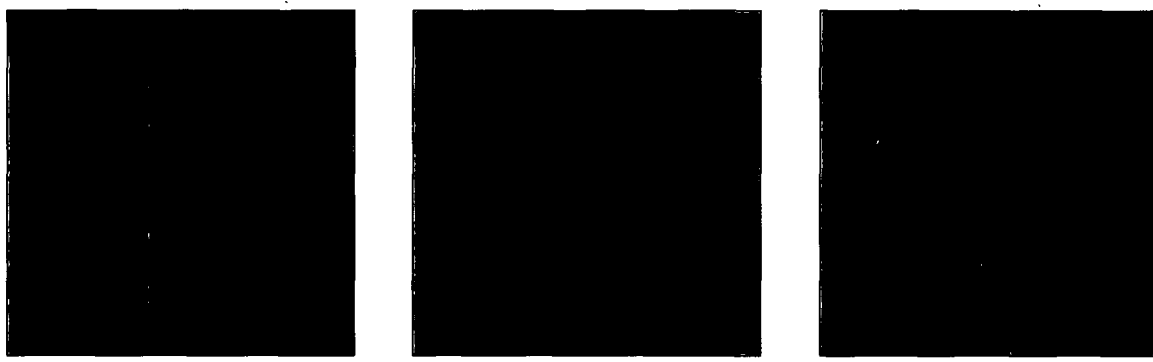


Figure A.7: *The "Lambertian Sphere" sequence. Image size is* 256×256.

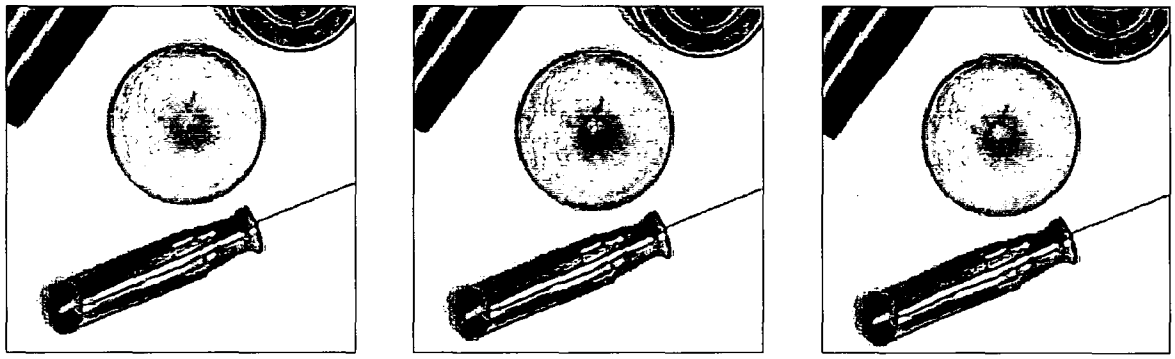
(a) (b) (c)

Figure A.8: *Frames 0, 16 and 32 from the "Strips" colour image test sequence. Image size is* 256×256.



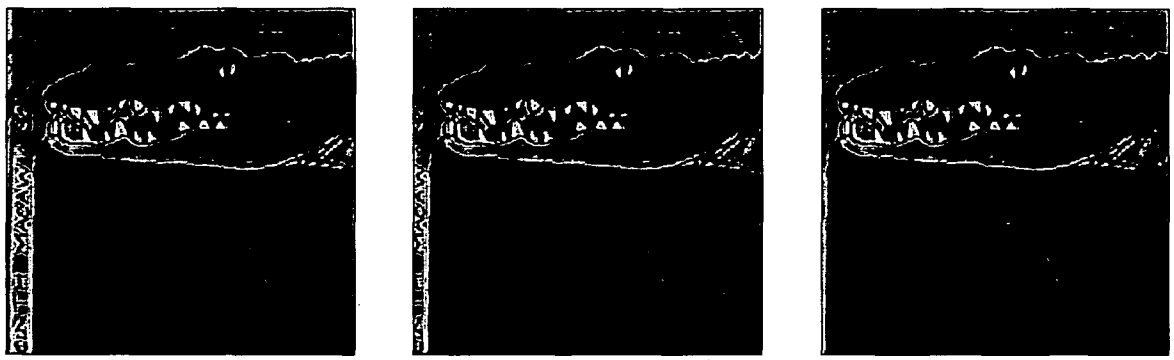
(a) (b) (c)

Figure A.9: *Frames 1, 5 and 9 from the colour shadow and "Isoluminance Test" image sequence. Image size is* 256×256.



(a) (b) (c)

Figure A.10: *Frames 1, 5 and 9 from the colour "Screwdriver" test sequence. Image size is* 512×512.



(a) (b) (c)

Figure A.11: *Frames 1, 5 and 9 from the colour "Poster" sequence. Image size is* 512×512.

A.2 Realistic Test Sequences

Five real test image sequences, and one realistic image sequence, were used in this work. These sequences were captured at a number of laboratories. Table A.2 lists the image sequences used.

Table A.2: Summary of real test images used.

Sequence	Comments and Speed Range
Yosemite Fly-Through	Created by Lynn Quam. Complex and challenging test case with occluding edges and severe aliasing in lower portion of images. Made available by Prof. John Barron. Email: barron@csd.uwo.ca, FTP: ftp.csd.uwo.ca 1 (clouds) to 4 (lower right) pixel/frame
Otte and Nagel's Block World	From Otte and Nagel[167], Made available by Prof. H-H. Nagel. Email: hhn@iitb.fhg.de HTTP://www.ira.uka.de less than 1.0 pixel/frame
Rubic Cube	From NASA-Ames Research Center. Diverging Image sequence. Also made available by John Barron. less than 1.0 pixel/frame
NASA Coke Can	From NASA-Ames Research Center. Diverging Image sequence. Also made available by John Barron. less than 1.0 pixel/frame
SRI Trees	From SRI corporation. Camera translates from left to right, viewing a grove of trees with a deep field of view. Also made available by John Barron. Velocities are as large as 2 pixel/frame
Hamburg Taxi	From Hamburg University, Prof. H-H. Nagel. 4 Independently moving objects. Also made available by John Barron. 0.3 to 1.0 pixel/frame

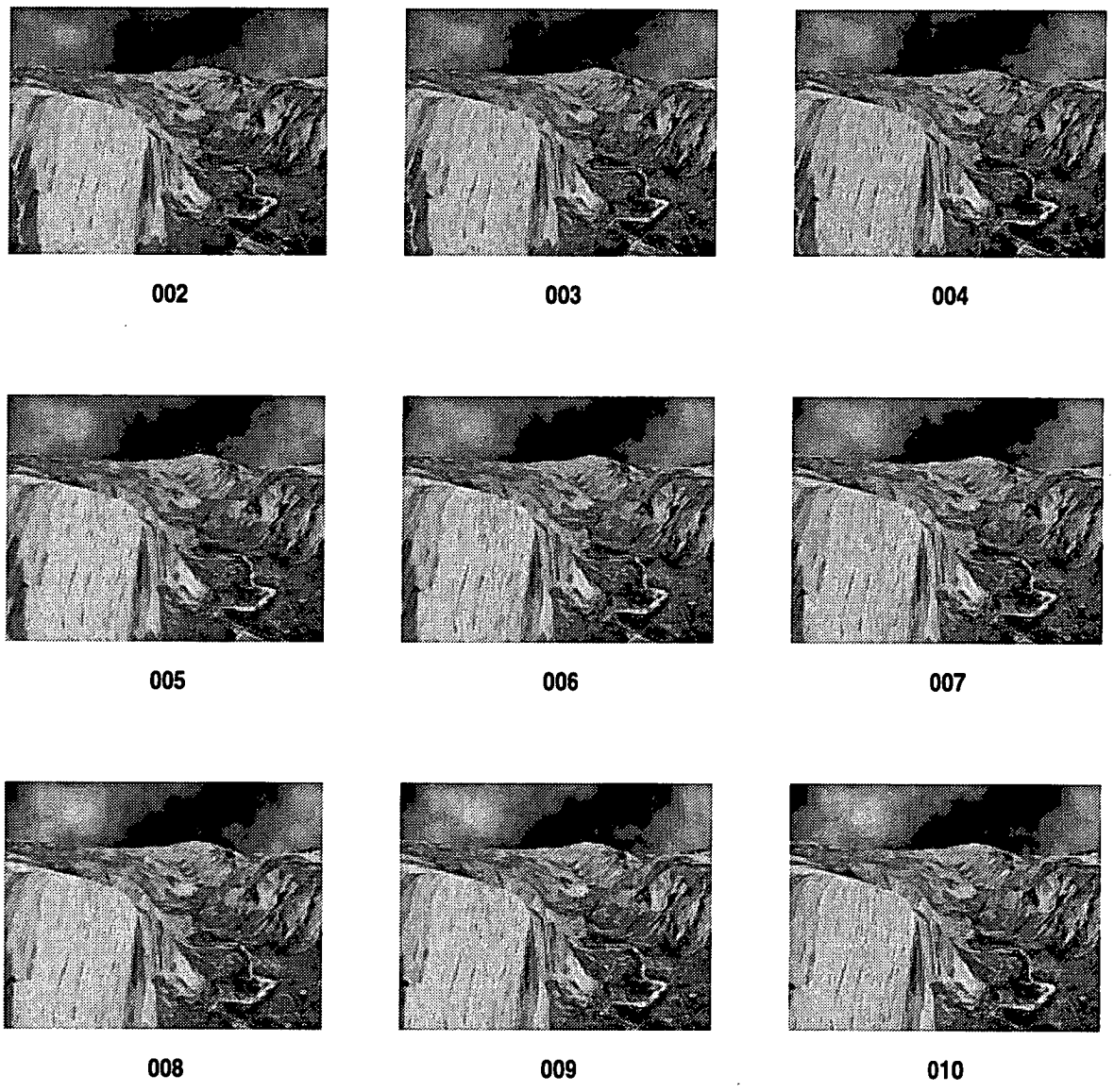


Figure A.12: *The "Yosemite Fly Through" sequence. Image size is* 316×252.

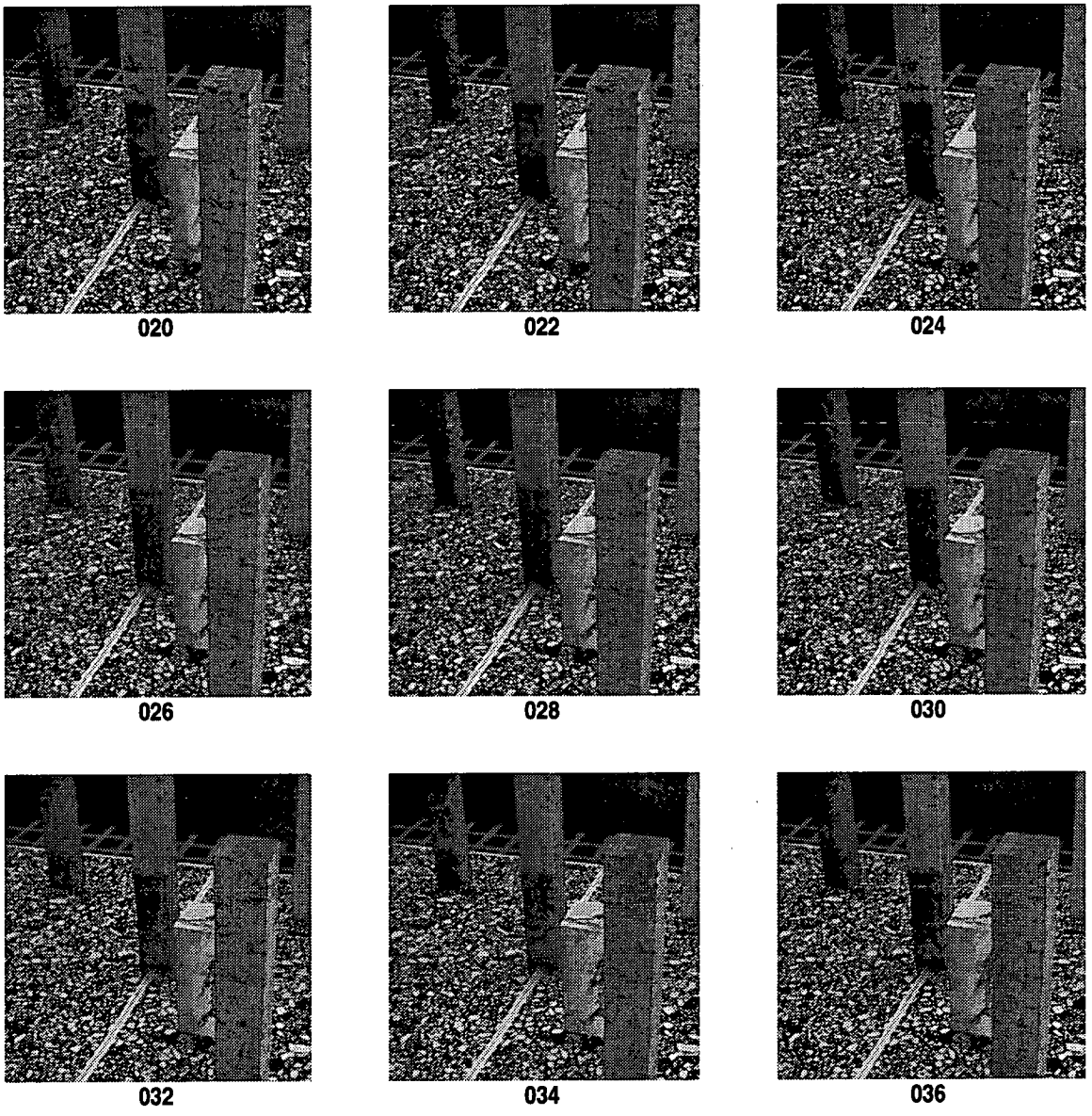


Figure A.13: *The "Otte and Nagel's Block World" sequence. Image size is* 512×512.

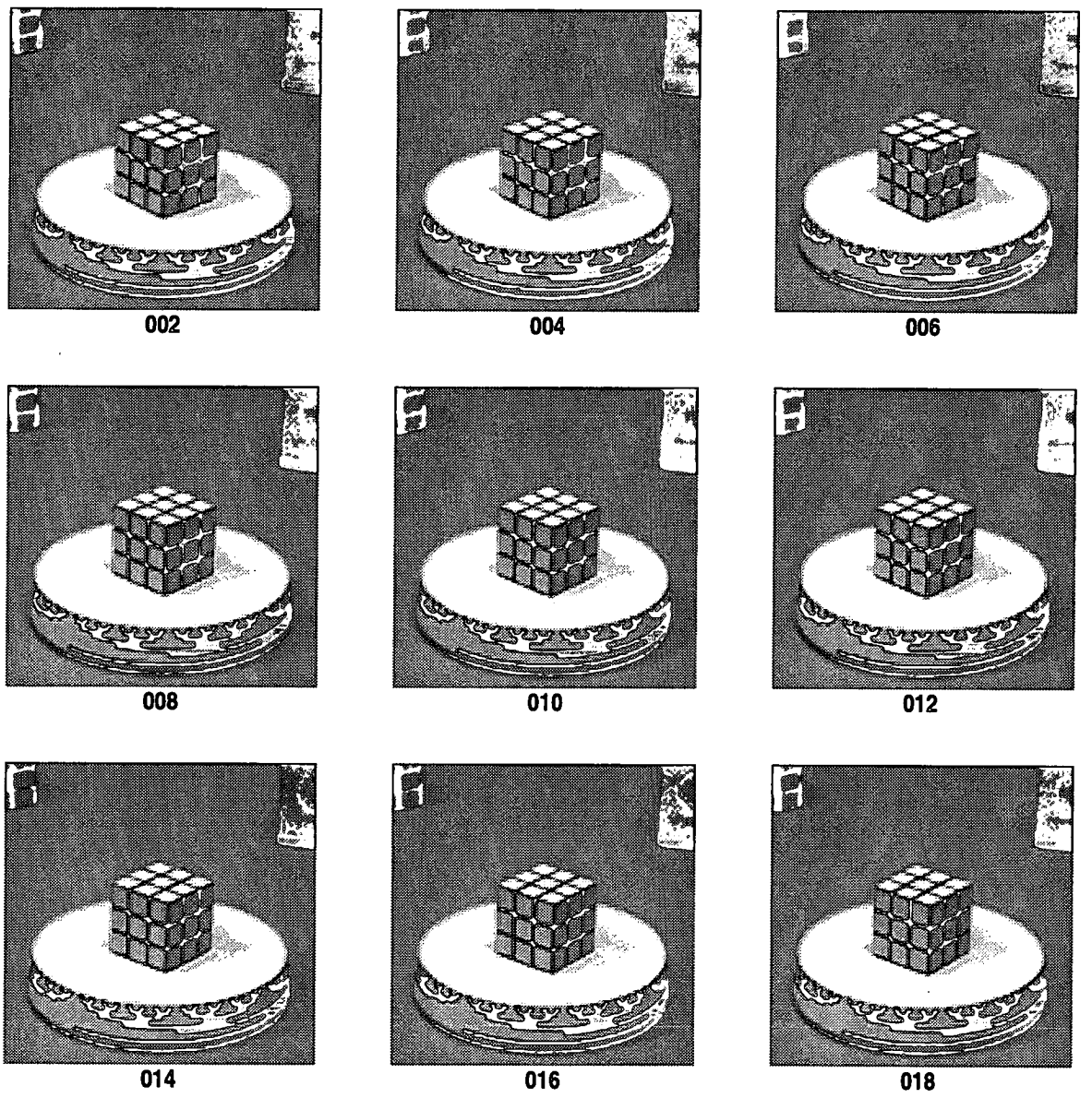


Figure A.14: *The "Rubic Cube" sequence. Image size is* 256×240.

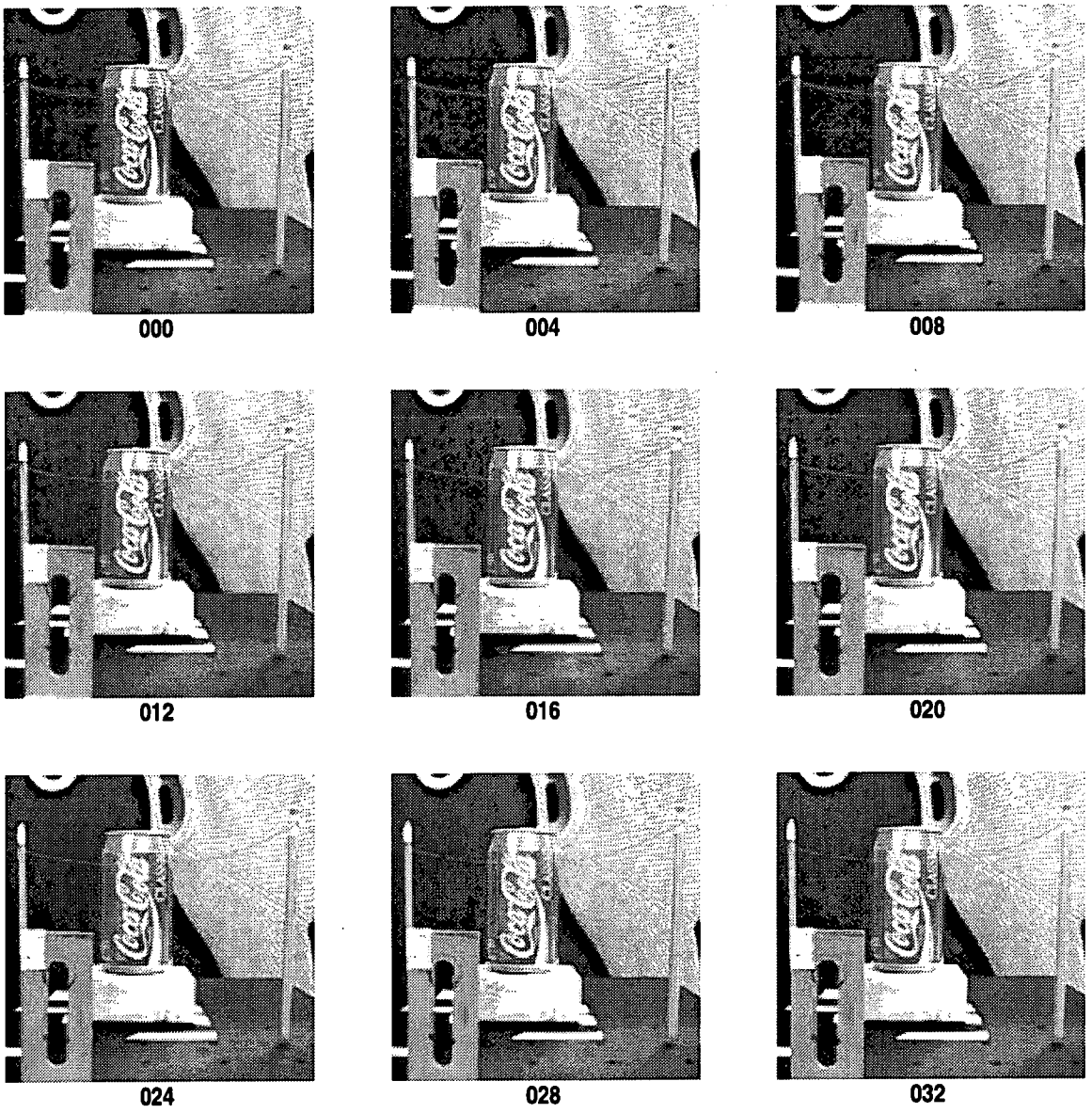


Figure A.15: *The "NASA Coke Can" sequence. Image size is* 300×300.

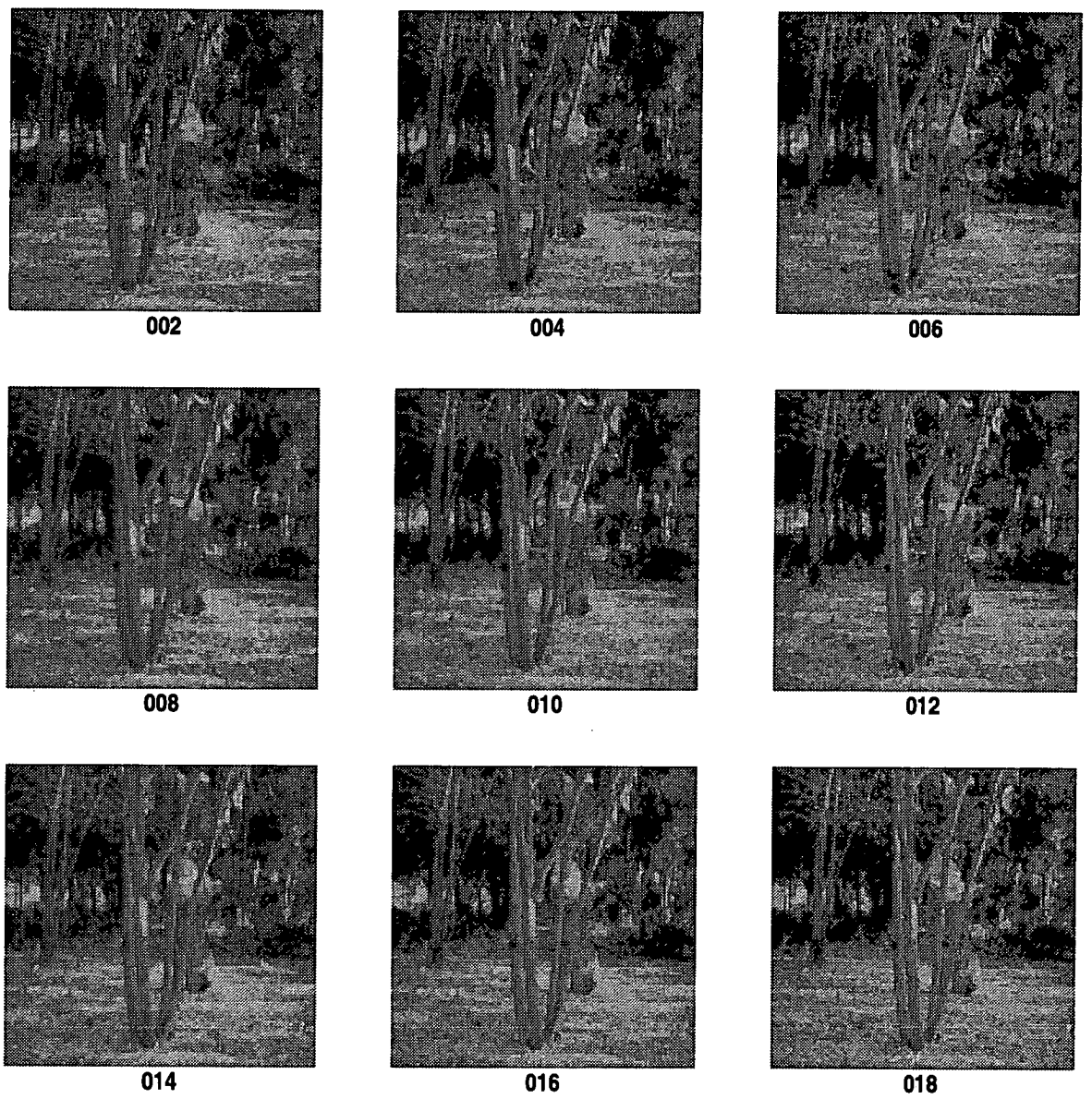


Figure A.16: *The "SRI Trees" sequence. Image size is* 256×233.

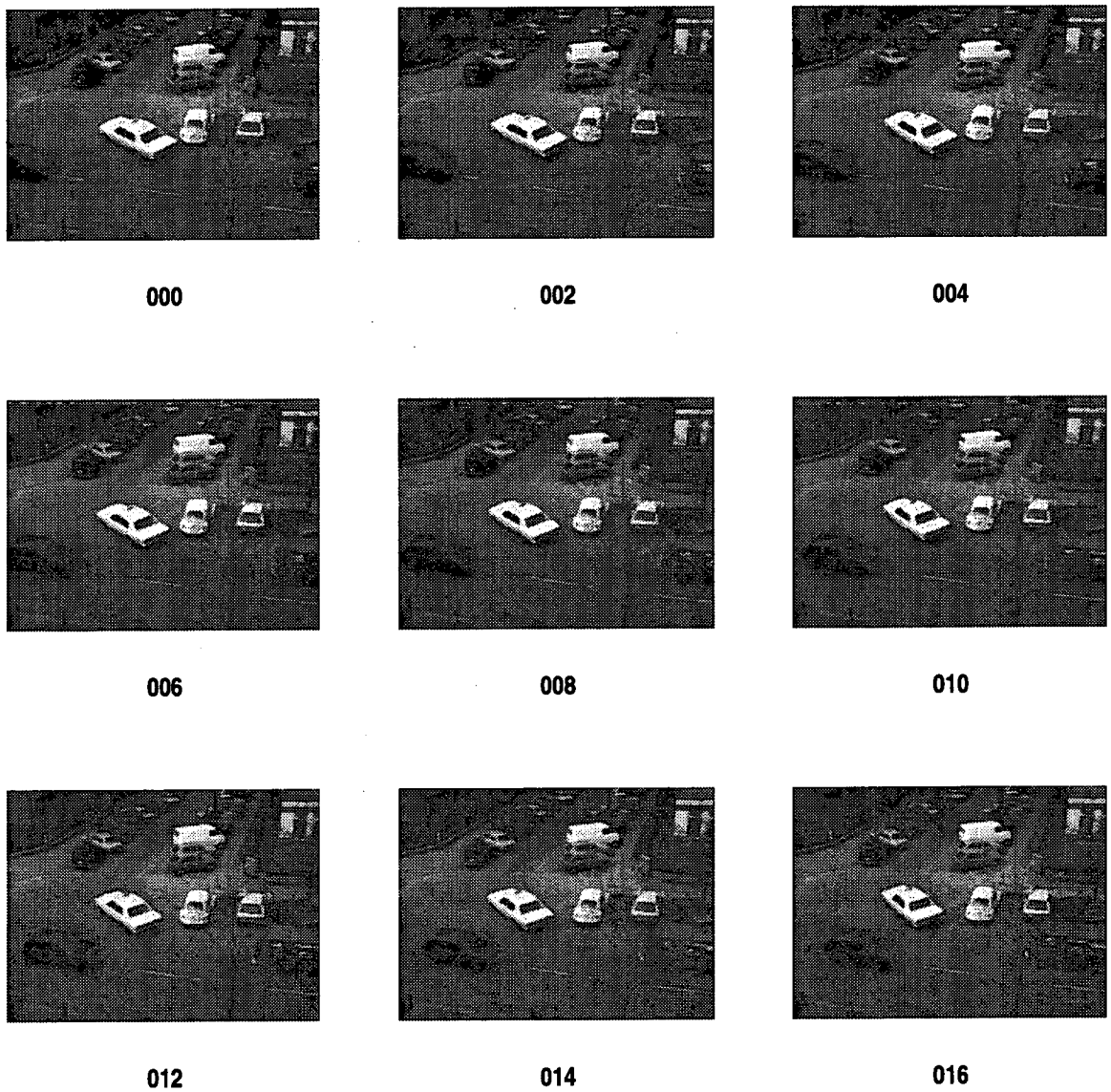


Figure A.17: *The "Hamburg Taxi" sequence. Image size is* 256×190.

Appendix B

Quaternion Algebra

Consider the following rotational transformations shown in Figure B.1, noting that positive rotations are given as an anti-clockwise turn looking into the axis of rotation.

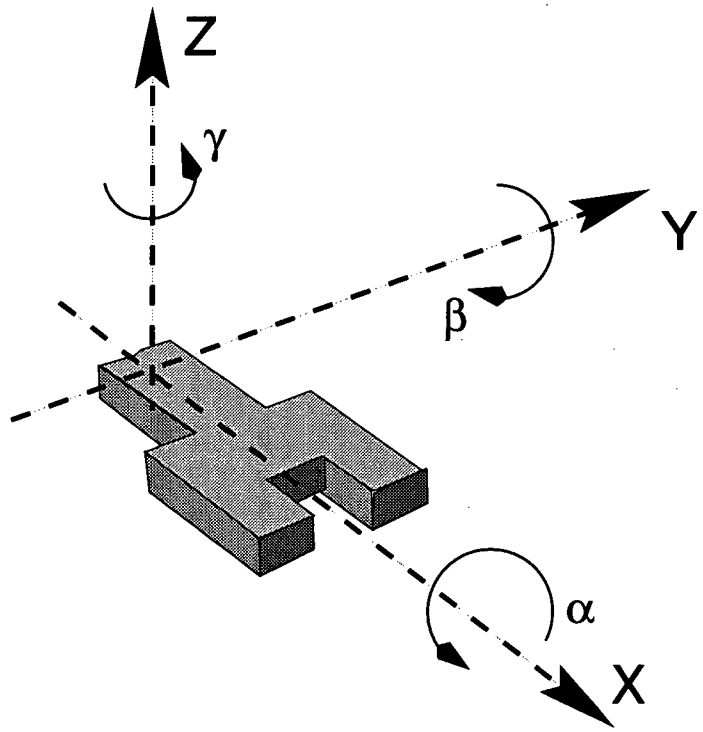


Figure B.1: *Roll (α), pitch (β) and yaw (γ) of a robot tool piece.*

For our purposes it is necessary to map every point in some input space x to some output space y undergoing a rotational transformation T such that,

$$y = T\{x\}. \tag{B.1}$$

This transformation can be expressed in a number of forms, we shall consider two representations: rotation matrices and quaternions.

B.1 Rotation Matrices

The most common representation of Eq.(B.1) is as a matrix-vector equation where T is a rotation matrix acting on the vector or point x. The derivation of the three rotation matrices for roll, pitch and yaw can be quickly and easily deduced.

Consider the case of roll, as shown in Figure B.2. Any point on a plane orthogonal to the roll axis will shift to a new point on that plane so as to maintain a constant distance or radius to the point where the axis of rotation intersects with the plane. All points located on the roll axis will remain stationary. By considering the orthogonal (x, y, z) components of each point individually the roll matrix is expressed by,

$$T_{\text{roll}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}. \tag{B.2a}$$

Similarly it can be shown that rotation matrices for pitch and yaw must be,

$$T_{\text{pitch}} = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}, \tag{B.2b}$$

$$T_{\text{yaw}} = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{B.2c}$$

Note that these transforms are formed on the basis that the rotation is positive if it is anti-clockwise when looking into the axis of rotation. If one is to use the convention of looking out of the axis of rotation, i.e. in the direction that the rotation axis points, then it is necessary to swap the sign of the angle.

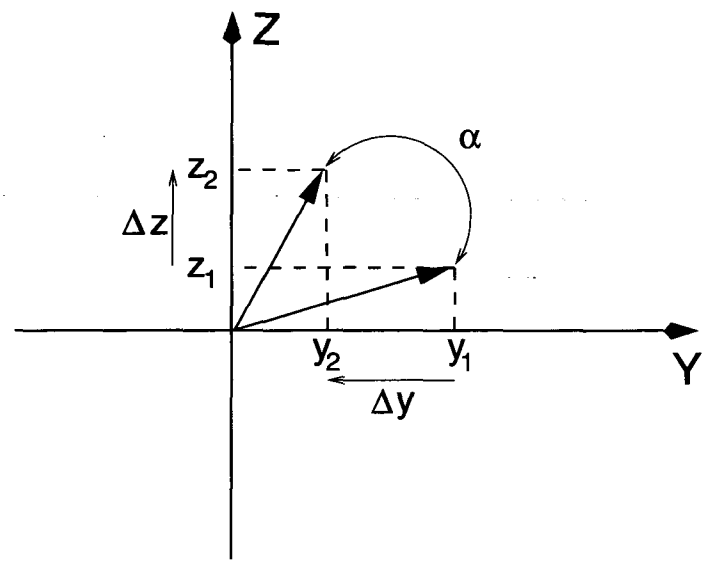


Figure B.2: *The point (x_1, y_1, z_1) transforms to the point (x_1, y_2, z_2) under the roll transform of angle α degrees.*

Composite rotations comprising combinations of pitch, roll, and yaw rotations, are formed by applying each transformation consecutively,

$$y = T_n T_{n-1} T_{n-2} ... T_1 x. \tag{B.3}$$

The order in which the rotations are applied is important as they are non-commutative; physically a pitch rotation followed by a roll rotation will not necessarily produce the same result as a roll rotation, of the same angle and direction, followed by a pitch rotation, of the same angle and direction. This point is well illustrated by the non-commutative property of matrix multiplication, i.e. $T_j T_k \neq T_k T_j$.

B.2 Quaternions

The quaternion is an alternative transform to the rotation matrix that has the same effect of transforming points in an input space to the output space by means of a rotation about a fixed rotation vector.

The quaternion was developed by Sir William Hamilton [97] at Trinity College Dublin during the mid 1800's in which he considered the quotient of two vectors. In Figure B.3(a) the quotient of two parallel vectors will yield a scalar value, which is the scale factor relating the two vectors. However, in the case of Figure B.3(b) it is clear that the quotient of the

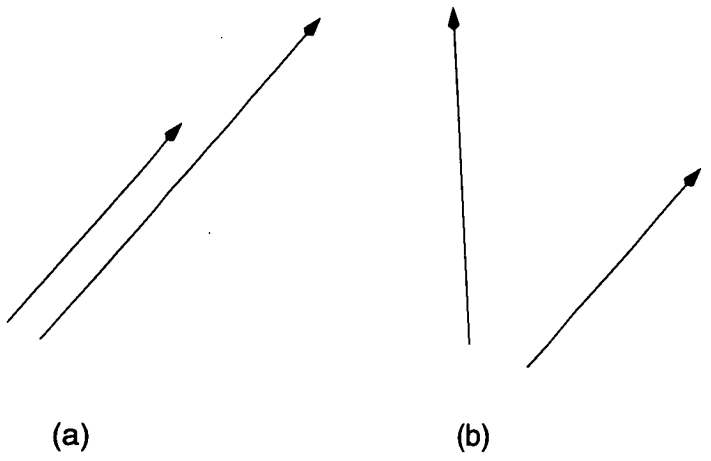


Figure B.3: *The relationship between two vectors (a) parallel, (b) defining a plane and angular separation.*

two non-parallel lines cannot be a simple scalar value relating the lengths of the two vectors, but must be a more complex object relating the relative angles of separation in a particular plane. Therefore a unit quaternion (one which has unit length) is a 4 parameter object: an angle and a plane of rotation represented by 3 parameters. Hamilton's quaternion is defined as a vector augmented with a scalar,

$$Q = \begin{bmatrix} q_s \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}. \tag{B.4}$$

Two operators, $S\{\}$ and $V\{\}$ can be defined to work on the quaternion Q. The operator $S\{\}$ returns the scalar part q_s and the operator $V\{\}$ returns the vector part (q_1, q_2, q_3). Degenerate forms of the quaternion are unambiguously represented by a scalar for when $V\{Q\} = 0$ and a vector when $S\{Q\} = 0$.

Hamilton developed a complete algebra for the quaternion, showing that quaternion addition,

$$Q_1 = Q_2 + Q_3 = \begin{bmatrix} S\{Q_2\} + S\{Q_3\} \\ V\{Q_2\} + V\{Q_3\} \end{bmatrix}, \tag{B.5}$$

is both commutative and associative.

Quaternion multiplication on the otherhand is only associative and not commutative,

$$Q_1 = Q_2 Q_3 = \begin{bmatrix} S\{Q_2\}S\{Q_3\} - V\{Q_2\} \cdot V\{Q_3\} \\ S\{Q_2\}V\{Q_3\} + S\{Q_3\}V\{Q_2\} + V\{Q_2\} \times V\{Q_3\} \end{bmatrix}, \tag{B.6}$$

because the cross-product term is not commutative. Quaternion multiplication forms a *closed group*. A closed group for an operator implies that the operator is associative, possesses an identity and has an inverse element belonging to the same set.

The identity quaternion is the scalar 1,

$$Q_I = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{B.7}$$

If we define the quaternion conjugate of Q to be,

$$Q^* = \begin{bmatrix} S\{Q\} \\ -V\{Q\} \end{bmatrix}, \tag{B.8}$$

then the quaternion norm is,

$$\|Q\|^2 = Q^* Q = S\{Q\}^2 + \|V\{Q\}\|^2, \tag{B.9}$$

i.e. the squared norm value is the sum of the squared elements. From Eq.(B.8) and Eq.(B.9) the identity element can be formed from any quaternion by employing its conjugate,

$$\frac{Q^* Q}{\|Q\|^2} = Q_I. \tag{B.10}$$

Therefore the inverse quaternion is,

$$Q^{-1} = \frac{1}{\|Q\|^2} Q^*. \tag{B.11}$$

Clearly for a unit quaternion, $Q^{-1} = Q^*$.

Other properties that readily follow include the commuted properties of conjugate and inverse products,

$$\{Q_1 Q_2\}^* = Q_2^* Q_1^*, \tag{B.12a}$$

$$\{Q_1 Q_2\}^{-1} = Q_2^{-1} Q_1^{-1}. \tag{B.12b}$$

By definition Hamilton defined the quaternion to be a quotient of two vectors. Therefore, the product of a quaternion and a vector in the plane of rotation necessarily results in a vector,

$$Q = a/b \qquad \Rightarrow \qquad Qb = a. \tag{B.13}$$

However, Spring[194] shows that Hamilton's definition of the quaternion is unsuitable if b does not lie wholly on the plane of rotation. In this case Hamilton's quaternion destroys the "symmetry" of b and in general a is a quaternion with a scalar component. He therefore defines the quaternion for a rotation ϕ about an axis n to be,

$$Q_{\phi,n} = \begin{bmatrix} \cos(\phi/2) \\ \sin(\phi/2) \cdot n \end{bmatrix}, \tag{B.14}$$

and that the rotation of points in the input space to the output space through angle ϕ as,

$$Q_{\phi,n}\, b\, Q^*_{\phi,n} = a. \tag{B.15}$$

Hamilton's quaternion by definition will only rotate a vector in the plane of rotation, while the definition given by Spring correctly rotates those components of b in the rotation plane leaving untouched those components orthogonal to it.

By applying the definition given in Eq.(B.14) we see that quaternions for a roll of α, pitch of β and yaw of γ are,

$$Q_{\text{roll}} = \begin{bmatrix} \cos(\alpha/2) \\ \sin(\alpha/2) \\ 0 \\ 0 \end{bmatrix}, \quad Q_{\text{pitch}} = \begin{bmatrix} \cos(\beta/2) \\ 0 \\ \sin(\beta/2) \\ 0 \end{bmatrix}, \quad Q_{\text{yaw}} = \begin{bmatrix} \cos(\gamma/2) \\ 0 \\ 0 \\ \sin(\gamma/2) \end{bmatrix}. \tag{B.16}$$

As in the case for composite rotations using rotation matrices, a single composite quaternion can be formed by successive quaternion multiplications,

$$\begin{aligned} y &= Q_n Q_{n-1} Q_{n-2} \ldots x \ldots Q^*_{n-2} Q^*_{n-1} Q^*_n \\ &= Q_n Q_{n-1} Q_{n-2} \ldots x (Q_n Q_{n-1} Q_{n-2} \ldots)^*. \end{aligned} \tag{B.17}$$

B.3 Concluding Remarks

First inspection of quaternion rotation reveals little advantage in its use over the popularly employed rotation matrices due to their considerable conceptual difficulty. However, once mastered a number of significant advantages can be noted:

- It has a simpler more compact notation.
- Less operations are required.
- Consists of 4 terms with 1 easily defined constraint as opposed to 9 terms and 6 constraint equations.
- The inverse is easily determined.

Bibliography

Books: Image Processing and Photogrammetry

[1] American Society of Photogrammetry. *Manual of Photogrammetry.* American Society of Photogrammetry, Washington DC., 1966.

[2] American Society of Photogrammetry. *Manual of Photographic Interpretation.* American Society of Photogrammetry, Washington DC., 1960.

[3] Dana H. Ballard and Christopher M. Brown. *Computer Vision.* Prentice-Hall, 1982.

[4] David J. Fleet. *Measurement of Image Velocity.* Kluwer-Academic Publishers, Dordrecht, The Netherlands, 1992.

[5] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing.* Addison-Wesley Publishing, 1992.

[6] Robert M. Haralick and Linda G. Shapiro. *Computer and Robot Vision.* Addison-Wesley Publishing Co., 1992.

[7] Berthold K.P. Horn. *Robot Vision.* MIT Press, Cambridge Mass., 3 edition, 1987.

[8] B. Jähne. *Spatio-Temporal Image Processing: Theory and Scientific Applications.* Springer-Verlag, Berlin, 1993.

[9] David Marr. *Vision.* W H Freeman, San Francisco, 1982.

[10] Francis H. Moffitt. *Photogrammetry.* International Textbook Company, 1967.

[11] David W. Murray and Bernard F. Buxton. *Experiments in the Machine Interpretation of Visual Motion.* MIT Press, London and Cambridge Mass., 1990.

[12] Jorge L.C. Sanz, editor. *Advances in Machine Vision.* Springer-Verlag, New York, 1988.

[13] Guy L. Scott. *Local and Global Interpretation of Moving Images.* Morgan Kaufmann Publishers, Inc., 1988.

[14] Stephen H. Spurr. *Photogrammetry and Photo Interpretation.* The Ronald Press Company, 1948.

[15] Shimon Ullman. *The Interpretation of Visual Motion.* MIT Press, Cambridge, MA., 1979.

Books: Mathematics and Least Squares Analysis

[16] A. Albert. *Regression and the Moore-Penrose Pseudoinverse.* Academic Press, 1972.

[17] H. Baher. *Analog and Digital Signal Processing.* John Wiley and Sons, Chichester, 1990.

[18] A Basilevsky. *Applied Matrix Algebra in the Statistical Sciences.* Elsevier Scientific Publishing Company, 1983.

[19] A. Ben-Israel and T.N.E. Greville. *Generalized Inverses: Theory and Applications.* John Wiley and Sons Limit, 1974.

[20] A. Bjerhammar. *Theory of Errors and Generalized Matrix Inverses.* Elsevier Scientific Publishing Company, 1973.

[21] Ronald Newbold Bracewell. *The Fourier Transform and its Applications.* McGraw-Hill International, 1978.

[22] Bruce W. Char and Others. *Maple V Library Reference Manual.* Springer-Verlag, New York, 1991.

[23] R.A. Fisher. *Contributions to Mathematical Statistics.* John Wiley and Sons Limit, 1950.

[24] D. Fletcher. *Unconstrained Optimization.* Wiley, New York, 1980.

[25] W.A. Fuller. *Measurement Error Models.* John Wiley and Sons Limit, 1987.

[26] A. Gelb, editor. *Applied Optimal Estimation.* MIT Press, Cambridge, Mass., 1988.

[27] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization.* Academic Press, 1981.

[28] E. Kreyszig. *Advanced Engineering Mathematics.* John Wiley and Sons Limit, Canada, sixth edition, 1988.

[29] Athanasios Papoulis. *The Fourier Integral and its Applications.* McGraw-Hill International, 1962.

[30] H.W. Sorenson. *Parameter Estimation: Principles and Problems.* Marcel Dekker, Inc., 1980.

Books: General

[31] Adobe Systems Incorprated. *PostScript Language Reference Manual.* Addison-Wesley Publishing Company, 1992.

[32] Brian D.O. Anderson and John B Moore. *Optimal Filtering.* Prentice-Hall International, 1979.

[33] Isaac Asimov. *Robot Dreams.* Byron Press Visual Publications Inc., 1986.

[34] William L. Brogan. *Modern Control Theory.* Prentice-Hall, New Jersey 07632, 1985.

[35] John Dunlop and D. Geoffrey Smith. *Telecommunications Engineering.* Chapman and Hall Publishing Company, 1984.

[36] Simon Haykin. *Communication Systems.* John Wiley and Sons Ltd., 1978.

[37] David H. Hubel. *Eye, Brain and Vision.* Scientific American Library, Washington, 1988.

[38] George Kennedy. *Electronic Communication Systems.* McGraw-Hill International., 1984.

[39] Y.H. Pao. *Adaptive Pattern Recognition and Neural Networks.* Addison-Wesley, 1989.

[40] Ian Sommerville. *Software Engineering.* Addison-Wesley Publishing Company, 3 edition, 1989.

[41] Sigvard Strandh. *A History of The Machine.* AB Nordbok, Gothenburg, Sweden, 1979.

[42] Herbert Taub and Donald L Schilling. *Principles of Communication Systems.* McGraw-Hill International., 1986.

Papers:

[43] Gilad Adiv. Inherent Ambiguities in Recoving 3-D Motion and Structure from a Noisy Flow Field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(5), pp. 477–489, May 1989.

[44] J.K. Aggarwal and N. Nandhakumar. On the Computation of Motion from Sequences of Images – A Review. *Proceedings of the IEEE*, **76**(8), pp. 917–935, August 1988.

[45] J.K. Aggarwal and C.H. Chien. *Advances in Machine Vision, Chapter:* 3-D Structures from 2-D Images, pages 64–121. Springer-Verlag, New York, 1988.

[46] Narendra Ahuja and A. Lynn Abbott. Active Stereo: Integrating Disparity, Vergence, Focus, Aperture and Calibration for Surface Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(10), pp. 1007–1029, October 1993.

[47] Janet Aisbett. Optical Flow with an Intensity-Weighted Smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(5), pp. 512–522, May 1989.

[48] Janet Aisbett. An Iterated Estimation of the Motion Parameters of a Rigid Body from Noisy Displacement Vectors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**(11), pp. 1092–1098, November 1990.

[49] John Aloimonos. Visual Shape Computation. *Proceedings of the IEEE*, **76**(8), pp. 899–916, August 1988.

[50] Stephen J. Anderson and David C. Burr. Spatial Summation Properties of Directionally Selective Mechanisms in Human Vision. *Journal of the Optical Society of America A*, **8**(8), pp. 1330–1339, August 1991.

[51] Stephen J. Anderson, David C. Burr, and M. Concetta Morrone. Two-Dimensional Spatial and Spatial-Frequencey Selectivity of Motion-Sensitive Mechanisms in Human Vision. *Journal of the Optical Society of America A*, **8**(8), pp. 1340–1351, August 1991.

[52] Andrew S.L. Bainbridge-Smith and Richard G. Lane. Determining Optical Flow Using a Differential Method. *Image and Vision Computing*, **15**(1), pp. 11–22, January 1997.

[53] John L. Barron, Allan D. Jepson, and John K. Tsotsos. The Feasibility of Motion and Structure from Noisy Time-Varying Image Velocity Information. *International Journal of Computer Vision*, **5**(3), pp. 239–269, 1990.

[54] John L. Barron, David J. Fleet, S.S. Beauchemin, and T.A. Burkitt. Performance of Optical Flow Techniques. *Technical Report:* Ref: TR-299, RPL-TR-9107, University of Western Ontario, London, Ontario, Canada N6A 5B7, July 1992.

[55] John L. Barron, David J. Fleet, S.S. Beauchemin, and T.A. Burkitt. Performance of Optical Flow Techniques. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, pages 236–242. International Pattern Recognition Society, June 1992.

[56] John L. Barron, David J. Fleet, and S.S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, **12**(1), pp. 43–77, 1994.

[57] John L. Barron and A. Liptay. Optical Flow to Measure Minute Increments in Plant Growth. *BioImaging*, 1994.

[58] Fredrik Bergholm and Stefan Carlsson. A "Theory" of Optical Flow. *CVGIP: Graphics Models and Image Processing*, **53**(2), pp. 171–188, 1991.

[59] Paul J. Besl. *Advances in Machine Vision, Chapter:* Active Optical Range Imaging Sensors, pages 1–63. Springer-Verlag, New York, 1988.

[60] Horst A. Beyer. Linejitter and geometric calibration of CCD-cameras. *ISPRS Journal of Photogrammetry and Remote Sensing*, **45**, pp. 17–32, 1990.

[61] Bik Bhanu and Wilhelm Burger. A Qualitative Approach to Dynamic Scene Understanding. *CVGIP: Image Understanding*, **54**(2), pp. 184–205, September 1991.

[62] Alberto Del Bimbo, Paolo Nesi, and Jorge L.C. Sanz. Analysis of Optical Flow Constraint. *IEEE Transactions on Image Processing*, **4**(4), pp. 460–469, April 1995.

[63] Patrick Bouthemy. A Maximum Likelihood Framework for Determining Moving Edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(5), pp. 499–511, May 1989.

[64] M.E. Bowman and A.K. Forrest. Transformation Calibration of a Camera Mounted on a Robot. *Image and Vision Computing*, **5**(4), pp. 261–266, November 1987.

[65] Ted J. Broida and Rama Chellappa. Estimation of Object Motion Parameters from Noisy Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(1), pp. 90–99, January 86.

[66] Ted J. Broida and Rama Chellappa. Performance Bounds for Estimating Three-Dimensional Motion Parameters from a Sequence of Noisy Images. *Journal of the Optical Society of America A*, **6**(6), pp. 879–889, June 1989.

[67] Peter J. Burt and Edward H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, **31**(4), pp. 532–540, April 1983.

[68] B.F. Buxton, H. Buxton, D.W. Murray, and N.S. Williams. *Advances in Artificial Intelligence, Chapter:* 3D Solutions to the Aperture Problem, pages 105–114. Elsevier Science Publisher B.V., North-Holland, 1985.

[69] Bernard F. Buxton and David W. Murray. Optical Flow Segmentation as an Ill-Posed and Maximum Likelihood Problem. *Image and Vision Computing*, **3**(4), pp. 163–169, November 1985.

[70] A.D. Calway and R. Wilson. Curve Extraction in Images Using a Multiresolution Framework. *CVGIP: Image Understanding*, **59**(3), pp. 359–366, May 1994.

[71] Marco Campani and Alessandro Verri. Motion Analysis from First-Order Properties of Optical Flow. *CVGIP: Image Understanding*, **56**(1), pp. 90–107, July 1992.

[72] John Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(6), pp. 679–698, November 1986.

[73] John Cardillo and Maher A. Sid-Ahmed. 3-D Position Sensing Using a Passive Monocular Vision System. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(8), pp. 809–813, August 1991.

[74] Patrick Cavanagh, Christopher W. Tyler, and Olga Eizner Favreau. Perceived velocity of moving chromatic gratings. *Journal of the Optical Society of America A*, **1**(8), pp. 893–898, August 1984.

[75] J.S. Chahl and M.V. Srinivasan. Visual Computation of Egomotion Using an Image Interpolation Technique. In *Proceedings ANZIIS-93*, pages 372–376, 1993.

[76] Homer H. Chen and Thomas S. Huang. Using Motion from Orthographic Views to Verify 3-D Point Matches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(9), pp. 872–878, September 1991.

[77] Francis Chin, Andrew Choi, and Yuhua Luo. Optimal Generating Kernals for Image Pyramids by Piecewise Fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(12), pp. 1190–1198, December 1992.

[78] Toshio M. Chin, William C. Karl, and Alan S. Willsky. Probabilistic and Sequential Computation of Optical Flow Using Temporal Coherence. *IEEE Transactions on Image Processing*, **3**(6), pp. 773–788, November 1994.

[79] C. Henry Chu and Edward J. Delp. Estimating Displacement Vectors from an Image Sequence. *Journal of the Optical Society of America A*, **6**(6), pp. 871–878, June 1989.

[80] Véronique Clément, Gérard Giraudon, Stéphane Houzelle, and Fadi Sandakly. Interpretation of Remotely Sensed Images in a Context of Multisensor Fusion Using a Multi-Specialist Architecture. *Technical Report:* No. 1768, Institut National de Recherche en Informatique et en Automatique, BP 93, F-06902 Sophia Antipolis Cedex, France, October 1992.

[81] R. Dennis Cook. Detection of Influential Observation in Linear Regression. *Technometrics*, **19**(1), pp. 15–18, February 1977.

[82] R. Dennis Cook and Sanford Weisberg. Characterizations of an Empirical Influence Function for Detecting Influential Cases in Regression. *Technometrics*, **22**(4), pp. 495–508, November 1980.

[83] Valérie Cornilleau-Pérès and Jacques Droulez. Velocity-Based Correspondence in Stereokinetic Images. *CVGIP: Image Understanding*, **58**(2), pp. 137–146, September 1993.

[84] Simon J. Cropper and David R. Badcock. Discriminating smooth from sampled motion: chromatic and luminance stimuli. *Journal of the Optical Society of America A*, **11**(2), pp. 515–530, February 1994.

[85] Subhodev Das and Narendra Ahuja. Performance Analysis of Stereo, Vergence and Focus as Depth Cues for Active Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(12), pp. 1213–1220, December 1995.

[86] John Ens and Peter Lawrence. An Investigation of Methods for Determining Depth from Focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(2), pp. 97–107, February 1993.

[87] Jia-Qi Fang and Thomas S. Huang. Some Experiments on Estimating the 3-D Motion Parameters of a Rigid Body from Two Consecutive Frames. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**(5), pp. 545–554, September 1984.

[88] David J. Fleet and Allan D. Jepson. Computation of Component Image Velocity from Local Phase Information. *International Journal of Computer Vision*, **5**(1), pp. 77–104, August 1990.

[89] David J. Fleet, Allan D. Jepson, and Michael R. M. Jenkin. Phase-Based Disparity Measurement. *CVGIP: Image Understanding*, **53**(2), pp. 198–210, March 1991.

[90] David J. Fleet and Keith Langley. Recursive Filters for Optical Flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(1), pp. 61–67, January 1995.

[91] Sergei V. Fogel. The Estimation of Velocity Vector Fields from Time-Varying Image-Sequences. *CVGIP: Image Understanding*, **53**(3), pp. 253–287, May 1991.

[92] David Forsyth and Andrew Zisserman. Reflections on Shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(7), pp. 671–679, July 1991.

[93] P.E. Gill and W. Murray. Algorithms for the Solution of the Nonlinear Least Squares Problem. *SIAM Journal Numerical Analysis*, **5**, pp. 977–992, October 1978.

[94] Helmut Glünder. Correlative Velocity Estimation: Visual Motion Analysis, Independent of Object Form, in Arrays of Velocity-Tuned Bilocal Detectors. *Journal of the Optical Society of America A*, **7**(2), pp. 255–263, February 1990.

[95] Andrei Gorea and Thomas V. Papathomas. Motion processing by chromatic and achromatic visual pathways. *Journal of the Optical Society of America A*, **6**(4), pp. 590–602, April 1989.

[96] Ardeshir Goshtasby, Fuhua Cheng, and Brian A. Barsky. B-Spline Curves and Surfaces Viewed as Digital Filters. *Computer Vision, Graphics, Image Processing*, **52**, pp. 264–275, 1990.

[97] W.R. Hamilton. *Elements of Quaternions.* Chelsea Publishing Company, 1899.

[98] K.J. Hanna and L. Tarassenko. Tracking Cataract by the 'Four-Line' method. *Image and Vision Computing*, **7**(1), pp. 57–62, February 1989.

[99] David J. Heeger. Model for the Extraction of Image Flow. *Journal of the Optical Society of America A*, **4**(8), pp. 1455–1471, August 1987.

[100] Robert J. Holt and Arun N. Netravali. Camera Calibration Problem: Some New Results. *CVGIP: Image Understanding*, **54**(3), pp. 368–383, November 1991.

[101] Berthold K.P. Horn. Hill Shading and the Reflectance Map. *Proceedings of the IEEE*, **69**(1), pp. 14–47, January 1981.

[102] Berthold K.P. Horn and Brian G. Schunck. Determining Optical Flow. *Artificial Intelligence*, **17**, pp. 185–203, 1981.

[103] Berthold K.P. Horn and John G. Harris. Rigid Body Motion from Range Image Sequences. *CVGIP: Image Understanding*, **53**(1), pp. 1–13, January 1991.

[104] Berthold K.P. Horn. Relative Orientation Revisited. *Journal of the Optical Society of America A*, **8**(10), pp. 1630–1638, October 1991.

[105] Berthold K.P. Horn and Brian G. Schunck. "Determining Optical Flow": a Retrospective. *Artificial Intelligence*, **59**, pp. 81–87, 1993.

[106] Robert Hummel and Venkataraman Sundareswaran. Motion Parameter Estimation from Global Flow Field Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(5), pp. 459–476, May 1993.

[107] Bobby R. Hunt. Lectures on Super-resolution. Lectures given at the University of Canterbury, New Zealand, August 1995.

[108] Katsushi Ikeuchi and Berthold K.P. Horn. Numerical Shape from Shading and Occulding Boundaries. *Artificial Intelligence*, **17**, pp. 141–184, 1981.

[109] Chris Bowman, Roger Clist, Olof Olsson, and Michael Rygol, editors. *Image and Vision Computing New Zealand Conference 1993.* Industrial Research Limited, Auckland, New Zealand, Aug 1993.

[110] Philip J. Bones, Graeme Coles, Michael Hayes, Peter Hilton, Wang Jian, Richard Lane, and Jimmy Li, editors. *Image and Vision Computing New Zealand Conference 1995.* Industrial Research Limited, Lincoln, New Zealand, Aug 1995.

[111] Lowell Jacobson and Harry Wechsler. Derivation of Optical Flow Using a Spatio-temporal-Frequency Approach. *Computer Vision, Graphics, Image Processing*, **38**, pp. 29–65, 1987.

[112] Ramesh C. Jain. Direct Computation of the Focus of Expansion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **5**(1), pp. 58–64, January 1983.

[113] Ramesh C. Jain. Segmentation of Frame Sequences Obtained by a Moving Observer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**(5), pp. 624–629, September 1984.

[114] Ray A. Jarvis. A Laser Time-of-Flight Range Scanner for Robotic Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **5**(5), pp. 505–512, September 1983.

[115] Ray A. Jarvis. Colour Edge Based Quad-Vision Ranging. In *Proceedings of Electronic Colour Imaging and Applications Workshop (DICTA-94)*, pages 83–88, Dec 1994.

[116] R. Jasinschi, A. Rosenfeld, and K. Sumi. Perceptual Motion Transparency: the Role of Geometrical Information. *Journal of the Optical Society of America A*, **9**(11), pp. 1865–1879, November 1992.

[117] Charles P. Jerian and Ramesh C. Jain. Structure from Motion — A Critical Analysis of Methods. *IEEE Transactions on Systems, Man. and Cybernetics*, **21**(3), pp. 572–587, May 1991.

[118] Michael R.M. Jenkin, Allen D. Jepson, and John K. Tsotsos. Techniques for Disparity Measurement. *CVGIP: Image Understanding*, **53**(1), pp. 14–30, January 1991.

[119] Dimitrios S. Kalivas and Alexander A. Sawchuk. A Region Matching Motion Estimation Algorithm. .*CVGIP: Image Understanding*, **54**(2), pp. 275–288, September 1991.

[120] Behrooz Kamgar-Parsi and Behzad Kamgar-Parsi. Evaluation of Quantization Error in Computer Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(9), pp. 929–940, September 1989.

[121] Kenichi Kanatani. Computational Projective Geometry. *CVGIP: Image Understanding*, **54**(3), pp. 333–348, November 1991.

[122] Kenichi Kanatani. Hypothesizing and Testing Geometric Properties of Image Data. *CVGIP: Image Understanding*, **54**(3), pp. 348–357, November 1991.

[123] Kenichi Kanatani. Unbiased Estimation and Statistical Analysis of 3-D Rigid Motion from Two Views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(1), pp. 37–50, January 1993.

[124] Joseph K. Kearney, William B. Thompson, and Daniel L. Boley. Optical Flow Estimation: An Error Analysis of Gradient-Based Methods with Local Optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(2), pp. 229–244, March 1987.

[125] Byungil Kim and Peter Burger. Depth and Shape from Shading Using the Photometric Stereo Method. *CVGIP: Image Understanding*, **54**(3), pp. 416–427, November 1991.

[126] H. Kirchner and H. Niemann. Finite Element Method for Determination of Optical Flow. *Pattern Recognition Letters*, **13**(2), pp. 131–141, February 1992.

[127] J. Kittler and J. Illingworth. Threshold Selection Based on a Simple Image Statistic. *Computer Vision, Graphics, Image Processing*, **30**, pp. 125–147, 1985.

[128] Jan J. Koenderink and Andrea J. van Doorn. The Internal Representation of Solid Shape with Respect to Vision. *Biological Cybernetics*, **32**, pp. 211–216, 1979.

[129] Jan J. Koenderink and Andrea J. van Doorn. Affine Structure from Motion. *Journal of the Optical Society of America A*, **8**(2), pp. 377–385, February 1991.

[130] Jan J. Koenderink and Andrea J. van Doorn. Second-Order Optical Flow. *Journal of the Optical Society of America A*, **9**(4), pp. 530–538, April 1992.

[131] Rakesh Kumar, Harprett S. Sawhney, and Allen R. Hanson. 3D Model Acquistion from Monocular Image Sequences. *Proceedings of the IEEE*, pages 209–215, 1992.

[132] Richard G. Lane. "Concerning your paper on the performance of optical flow methods." Private email correspondence with J.L. Barron, February 1994.

[133] Chia-Hoang Lee and Azriel Rosenfeld. Improved Methods of Estimating Shape from Shading Using the Light Source Coordinate System. *Artificial Intelligence*, **26**, pp. 125–143, 1985.

[134] Chia-Hoang Lee. Time-Varying Images: The Effect of Finite Resolution on Uniqueness. *CVGIP: Image Understanding*, **54**(3), pp. 325–332, November 1991.

[135] Dah Jye Lee, Sunanda Mitra, and Thomas F. Krile. Analysis of Sequential Complex Images, Using Feature Extraction and Two-Dimensional Cepstrum Techniques. *Journal of the Optical Society of America A*, **6**(6), pp. 863–870, June 1989.

[136] Reimar Lenz and Dieter Fritsch. Accuracy of Videometry with CCD sensors. *ISPRS Journal of Photogrammetry and Remote Sensing*, **45**, pp. 90–110, 1990.

[137] Lingxiao Li and James H. Duncan. 3-D Translational Motion and Structure from Binocular Image Flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(7), pp. 657–667, July 1993.

[138] Ping Liang and John S. Todhunter. Representation and Recognition of Surface Shapes in Range Images: A Differential Geometry Approach. *Computer Vision, Graphics, Image Processing*, **52**, pp. 78–109, 1990.

[139] Shih-Ping Liou and Ramesh C. Jain. An Approach to Three-Dimensional Image Segmentation. *CVGIP: Image Understanding*, **53**(3), pp. 237–252, May 1991.

[140] T. Lin and John L. Barron. Image Reconstruction Error for Optical Flow. *Technical Report:* Ref: TR-412, University of Western Ontario, London, Ontario, Canada N6A 5B7, February 1994.

[141] Seppo Linnainmaa, David Harwood, and Larry S. Davis. Pose Determination of a Three-Dimensional Object Using Triangle Pairs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**(5), pp. 634–647, September 1988.

[142] H.C. Longuet-Higgins. A Computer Algorithm for Reconstructing a Scene from Two Projections. *Nature*, **293**, pp. 133–135, September 1981.

[143] Bruce D. Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Proceedings of DARPA Image Understanding Workshop*, pages 121–130, 1981.

[144] Song De Ma. A Self-Calibration Technique for Active Vision Systems. *IEEE Transactions on Robotics and Automation*, **12**(1), pp. 114–120, February 1996.

[145] A.K. Mackworth. Interpreting Pictures of Polyhedral Scenes. *Artificial Intelligence*, **4**, pp. 121–137, 1973.

[146] Philip F. McLauchlan and David W. Murray. Active Camera Calibration for a Head-Eye Platform Using the Variable State-Dimension Filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**(1), pp. 15–22, January 1996.

[147] D. Marr and E. Hildreth. Theory of Edge Detection. *Proceedings of the Royal Society of London B*, **207**, pp. 187–217, 1980.

[148] Jasna Maver and Ruzena Bajcsy. Occlusions as a Guide for Planning the Next View. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(5), pp. 417–433, May 1993.

[149] S.J. Maybank. Algorithm for Analysing Optical Flow Based on the Least-Squares Method. *Image and Vision Computing*, **4**(1), pp. 38–42, February 1986.

[150] Andrew B. Metha, Algis J. Vingrys, and David R. Badcock. Detection and Discrimination of Moving Stimuli: The Effects of Color, Luminance, and Eccentricity. *Journal of the Optical Society of America A*, **11**(6), pp. 1697–1709, June 1994.

[151] A. Zvi Meiri. On Monocular Perception of 3-D Moving Objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**(6), pp. 582–583, November 1980.

[152] Enrico De Micheli, Vincent Torre, and Sergio Uras. The Accuracy of the Computation of Optical Flow and of the Recovery of Motion Parameters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(5), pp. 434–447, May 1993.

[153] David W. Murray and Bernard F. Buxton. Scene Segmentation from Visual Motion Using Global Optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(2), pp. 220–228, March 1987.

[154] Hans-Hellmut Nagel. Representation of Moving Rigid Objects Based on Visual Observations. *IEEE Computer Magazine*, pages 29–39, August 1981.

[155] Hans-Hellmut Nagel. Displacement Vectors Derived from Second-Order Intensity Variations in Image Sequences. *Computer Vision, Graphics, Image Processing*, **21**, pp. 85–117, 1983.

[156] Hans-Hellmut Nagel. An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(5), pp. 565–593, September 1986.

[157] Hans-Hellmut Nagel. On the Estimation of Optical Flow: Relations between Different Approaches and Some New Results. *Artificial Intelligence*, **33**, pp. 299–324, 1987.

[158] Hans-Hellmut Nagel. On a Constraint Equation for the Estimation of Displacement Rates in Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(1), pp. 13–30, January 1989.

[159] Nasser M. Nasrabadi, Sandra P. Clifford, and Yi Liu. Integration of Stereo Vision and Optical Flow by Using an Energy-Minimization Approach. *Journal of the Optical Society of America A*, **6**(6), pp. 900–907, June 1989.

[160] Shree K. Nayar, Katsushi Ikeuchi, and Takeo Kanade. Surface Reflection: Physical and Geometrical Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(7), pp. 611–634, July 1991.

[161] Shahriar Negahdaripour. Closed-Form Relationship Between the Two Interpretations of a Moving Plane. *Journal of the Optical Society of America A*, **7**(2), pp. 279–285, February 1990.

[162] Randal C. Nelson and Ramprasad Polana. Qualitative Recognition of Motion Using Temporal Texture. *CVGIP: Image Understanding*, **56**(1), pp. 78–89, July 1992.

[163] Paolo Nesi. Variational Approach to Optical Flow Estimation Managing Discontinuities. *Image and Vision Computing*, **11**(7), pp. 419–439, September 1993.

[164] H. Nicolas and C. Labit. Motion and Illumination Variation Estimation Using a Hierarchy of Models: Application to Image Sequence Coding. *Technical Report:* No: 742, Institut de Recherche en Informatique et Systèmes Aléatories, Campus de Beaulieu – 35042 Rennes Cedex, France, June 1993.

[165] Atsushi Nomura and Hidetoshi Miike. Field Theory Approach for Determining Optical Flow. *Pattern Recognition Letters*, **12**(3), pp. 183–190, March 1991.

[166] Masami Ogata and Takao Sato. Motion-Detection Model with Two Stages: Spatiotemporal Filtering and Feature Matching. *Journal of the Optical Society of America A*, **9**(3), pp. 377–387, March 1992.

[167] M. Otte and H.-H. Nagel. Optical Flow Estimation: Advances and Comparisons. In Jan-Olof Eklundh, editor, *Computer Vision — ECCV'94*, pages 51–60. Springer-Verlag, May 1994.

[168] Alex Paul Pentland. Local Shading Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**(2), pp. 170–187, March 1984.

[169] Alex Paul Pentland. Shading into Texture. *Artificial Intelligence*, **29**, pp. 147–170, 1986.

[170] Alex Paul Pentland. A New Sense for Depth of Field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(4), pp. 523–531, July 1987.

[171] Alex Paul Pentland. Photometric Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(9), pp. 879–890, September 1991.

[172] John A. Perrone. Simple Technique for Optical Flow Estimation. *Journal of the Optical Society of America A*, **7**(2), pp. 264–278, February 1990.

[173] John A. Perrone. Model for the Computation of Self-Motion in Biological Systems. *Journal of the Optical Society of America A*, **9**(2), pp. 177–194, February 1992.

[174] D.L.D. Phillips, Q.X. Wu, and J. McGregor. Cloud Motion Vectors From Sequential Satellite Images. In *Proceedings of the First Conference of Image and Vision Computing New Zealand*, pages 291–297, 1993.

[175] Keith Price. Anything You Can Do, I Can Do Better (No You Can't)... *Computer Vision, Graphics, Image Processing*, **36**, pp. 387–391, 1986.

[176] P. Puget and T. Skordas. Calibrating a Mobile Camera. *Image and Vision Computing*, **8**(4), pp. 341348, November 1990.

[177] Daniel Raviv and James S. Albus. A Closed-Form Massively-Parallel Range-from-Image-Flow Algorithm. *IEEE Transactions on Systems, Man. and Cybernetics*, **22**(2), pp. 322–327, March 1992.

[178] P. Remagnino, J. Illingworth, J. Kittler, and J. Matas. Intentional Control of Camera Look Direction and Viewpoint in an active Vision System. *Image and Vision Computing*, **13**(2), pp. 79–87, March 1995.

[179] Harpreet S. Sawhney, John Oliensis, and Allen R. Hanson. Image Description and 3-D Reconstruction from Image Trajectories of Rotational Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(9), pp. 885–898, September 1993.

[180] M. Ibrahim Sezan and Reginald L. Lagendijk, editors. *Motion Analysis and Image Sequence Processing.* Kluwer Academic Publishers, 1993.

[181] Brian G. Schunck. The Image Flow Constraint Equation. *Computer Vision, Graphics, Image Processing*, **35**, pp. 20–46, 1986.

[182] Brian G. Schunck. Image Flow Segmentation and Estimation by Constraint Line Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(10), pp. 1010–1027, October 1989.

[183] Guy L. Scott. 'Four-Line' Method of Locally Estimating Optical Flow. *Image and Vision Computing*, **5**(2), pp. 67–72, May 1987.

[184] T.Y. Shih. An Investigation into the Computational Procedures of Space Resection by Collinearity Equations. *Photogrammetric Record*, **13**(5), pp. 433–440, April 1990.

[185] Eero P. Simoncelli, Edward H. Adelson, and David J. Heeger. Probability Distributions of Optical Flow. *Proceedings of the IEEE*, pages 310–315, 1991.

[186] Ajit Singh and Peter Allen. Image-Flow Computation: An Estimation-Theoretic Framework and a Unified Perspective. *CVGIP: Image Understanding*, **56**(2), pp. 152–177, September 1992.

[187] A.T. Smith and G.K. Edgar. Perceived Speed and Direction of Complex Gratings and Plaids. *Journal of the Optical Society of America A*, **8**(7), pp. 1161–1171, July 1991.

[188] A.T. Smith, R.F. Hess, and C.L. Baker, Jr. Direction Identification Thresholds for Second-Order Motion in Central and Peripheral Vision. *Journal of the Optical Society of America A*, **11**(2), pp. 506–514, February 1994.

[189] Phillip Smith, Banavar Sridhar, and Bassam Hussien. Vision-Based Range Estimation Using Helicopter Flight Data. *Proceedings of the IEEE*, pages 202–208, 1992.

[190] M.A. Snyder. On the Mathematical Foundations of Smoothness Constraints for the Determination of Optical Flow and for Surface Reconstructions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(11), pp. 1105–1114, November 1991.

[191] P.J. Sobey and M.V. Srinivasan. Measurement of Optical Flow by a Generalized Gradient Scheme. *Journal of the Optical Society of America A*, **8**(9), pp. 1488–1498, September 1991.

[192] P.J. Sobey, M.G. Nagle, Y.V. Venkatesh, and M.V. Srinivasan. Measurement of Complex Optical Flow with use of an Augmented Generalized Gradient Scheme. *Journal of the Optical Society of America A*, **11**(11), pp. 2787–2798, November 1994.

[193] L.A. Spacek. Edge Detection and Motion Detection. *Image and Vision Computing*, **4**(1), pp. 43–56, 1986.

[194] Kerry W. Spring. Euler Parameters and the use of Quaternion Algebra in the Manipulation of Finite Rotations: A Review. *Mechanism and Machine Theory*, **21**(5), pp. 365–373, 1986.

[195] M.V. Srinivasan. An Image-interpolation Technique for the Computation of 2-D Motion. In *Proceedings ANZIIS-93*, pages 367–371, 1993.

[196] Thomas M. Strat and Martin A. Fischler. One-Eyed Stereo: A General Approach to Modeling 3-D Scene Geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(6), pp. 730–741, November 1986.

[197] Venkataraman Sundareswaran. *Global Methods for Image Motion Analysis.* PhD thesis, New York University, October 1992.

[198] Demetri Terzopoulos. Image Analysis Using Multigrid Relaxation Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(2), pp. 129–139, March 1986.

[199] Demetri Terzopoulos. Regularization of Inverse Visual Problems Involving Discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(4), pp. 413–424, July 1986.

[200] D.W. Thompson and J.L. Mundy. Three-Dimensional Model Matching from an Unconstrained Viewpoint. *Proceedings of IEEE Robotics and Automation*, pages 208–220, 1987.

[201] William B. Thompson, Kathleen M. Mutch, and Valdis A. Berzins. Dynamic Occlusion Analysis in Optical Flow Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **7**(4), pp. 374–383, July 1985.

[202] William B. Thompson and James S. Painter. Qualitative Constraints for Structure-from-Motion. *CVGIP: Image Understanding*, **56**(1), pp. 69–77, July 1992.

[203] Carlo Tomasi and Takeo Kanade. Shape and Motion from Image Streams: a Factorization Method *Chapter:* 2. Point Features in 3D Motion. *Technical Report:* Ref: CMU-CS-91-105, Carnegie Mellon University, Pittsburgh, PA 15213, USA, January 1991.

[204] Carlo Tomasi and Takeo Kanade. Shape and Motion from Image Streams: a Factorization Method *Chapter:* 3. Detection and Tracking of Point Features. , Carnegie Mellon University, Pittsburgh, PA 15213, USA, 1991.

[205] P.H.S. Torr and D.W. Murray. Statistical Detection of Independent Movement from a Moving Camera. *Image and Vision Computing*, **11**(4), pp. 180–187, May 1993.

[206] Vincent Torre and Thomaso A. Poggio. On Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(2), pp. 147–162, March 1986.

[207] Oleh Tretiak and Louis Pastor. Velocity Estimation from Image Sequences with Second Order Differential Operators. In *Proceedings International Conference on Pattern Recognition*, pages 16–19, 1984.

[208] Tom Troscianko and Manfred Fahle. Why do isoluminant stimuli appear slower? *Journal of the Optical Society of America A*, **5**(6), pp. 871–880, June 1988.

[209] Shimon Ullman. Analysis of Visual Motion by Biological and Computer Systems. *IEEE Computer Magazine*, pages 57–69, August 1981.

[210] Michael Unser, Akram Aldroubi, and Murray Eden. Recursive Regularization Filters: Design, Properties, and Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(3), pp. 272–277, March 1991.

[211] Michael Unser, Akram Aldroubi, and Murray Eden. Fast B-Spline Transforms for Continuous Image Representation and Interpolation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(3), pp. 277–285, March 1991.

[212] Alessandro Verri and Tomaso Poggio. Motion Field and Optical Flow: Qualitative Properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(5), pp. 490–498, May 1989.

[213] Alessandro Verri, F. Girosi, and Vincent Torre. Mathematical Properties of the Two-Dimensional Motion Field: From Singular Points to Motion Parameters. *Journal of the Optical Society of America A*, **6**(5), pp. 698–712, May 1989.

[214] Alessandro Verri, F. Girosi, and Vincent Torre. Differential Techniques for Optical Flow. *Journal of the Optical Society of America A*, **7**(5), pp. 912–922, May 1990.

[215] Ching-Cheng Wang. Extrinsic Calibration of a Vision Sensor Mounted on a Robot. *IEEE Transactions on Robotics and Automation*, **8**(2), pp. 161–175, April 1992.

[216] William H. Warren Jr. and Danial J. Hannon. Eye Movements and Optical Flow. *Journal of the Optical Society of America A*, **7**(1), pp. 160–169, January 1990.

[217] Andrew B. Watson and Michael P. Eckert. Motion-Contrast Sensitivity: Visibility of Motion Gradients of Various Spatial Frequencies. *Journal of the Optical Society of America A*, **11**(2), pp. 496–505, February 1994.

[218] Allen M. Waxman and Sarvajit S. Sinha. Dynamic Stereo: Passive Ranging to Moving Objects from Relative Image Flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(4), pp. 406–412, July 1986.

[219] Allen M. Waxman and James H. Duncan. Binocular Image Flows: Steps Toward Stereo-Motion Fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(6), pp. 715–412, November 1986.

[220] Juyang Weng, Thomas S. Huang, and Narendra Ahuja. 3-D Motion Estimation, Understanding, and Prediction from Noisy Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(3), pp. 370–389, May 1987.

[221] Juyang Weng, Thomas S. Huang, and Narendra Ahuja. Motion and Structure from Two Perspective Views: Algorithms, Error Analysis, and Error Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(5), pp. 451–426, May 1989.

[222] Juyang Weng, Narendra Ahuja, and Thomas S. Huang. Optimal Motion and Structure Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(9), pp. 864–884, September 1993.

[223] Peter Whaite and Frank P. Ferrie. From Uncertainty to Visual Exploration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(10), pp. 1038–1049, October 1991.

[224] Charles S. Wiles. *Closing the Loop on Multiple Motions.* PhD thesis, University of Oxford, 1995.

[225] Donna J. Williams and Mubarak Shah. Edge Contours Using Multiple Scales. *Computer Vision, Graphics, Image Processing*, **51**, pp. 256–274, 1990.

[226] Darryl Willick and Yee-Hong Yang. Experimental Evaluation of Motion Constraint Equations. *CVGIP: Image Understanding*, **54**(2), pp. 206–214, September 1991.

[227] R. Wilson and A.H. Bhalerao. Kernal Designs for Efficient Multiresolution Edge Detection and Orientation Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(3), pp. 384–390, March 1992.

[228] Kwang Yoen Wohn, Jian Wu, and Roger W. Brockett. A Contour-Based Recovery of Image Flow: Iterative Transformation Method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(8), pp. 746–760, August 1991.

[229] Jian Wu and Kwang Yoen Wohn. On the Deformation of Image Intensity and Zero-Crossing Contours under Motion. *CVGIP: Image Understanding*, **53**(1), pp. 66–75, January 1991.

[230] Q.X. Wu, D. Pairman, S. McNeil, and E.J. Barnes. Computing Advective Velocities from Satellite Images of Sea Surface Temperature. *IEEE Transactions on Geoscience and Remote Sensing*, **30**(1), pp. 166–176, 1992.

[231] Yoshio Yasumoto and Gerard Medioni. Robust Estimation of Three-Dimensional Motion Parameters from a Sequence of Image Frames Using Regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(4), pp. 464–471, July 1986.

[232] Gem-Sun Jason Young and Rama Chellappa. Statistical Analysis of Inherent Ambiguities in Recovering 3-D Motion from a Noisy Flow Field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(10), pp. 995–1013, October 1992.

[233] Zhengyou Zhang, Quang-Tuan Luong, and Olivier Faugeras. Motion of an Uncalibrated Stereo Rig: Self-Calibration and Metric Reconstruction. *IEEE Transactions on Robotics and Automation*, **12**(1), pp. 103–113, February 1996.

[234] Xinhua Zhuang. A Simplification to Linear Two-View Motion Algorithms. *Computer Vision, Graphics, Image Processing*, **46**, pp. 175–178, 1989.